Kobunsha

PC/AT互換機対応

CONFIG.SYSがわかる本

宮田　茂

CONFIG.SYSとAUTOEXEC.BATの

修・正・加・工・自・由・自・在

以下のパソコンをお持ちの方は必読の一冊

IBM、エイサー、エプソンダイレクト、コンパック、デル、
DEC、東芝、パナソニック、富士通などのパソコンは、
PC/AT互換機です。いわゆるDOS/Vパソコンです。

自分のハードに最適な環境設定を実現する

DOS/Vパソコンを快適に利用していこうとするなら、日本語を利用するためのソフトウェアの設定や、周辺機器を利用するための設定、あるいはメモリのカスタマイズといった作業が必須です。

本書では、これらすべてを「環境設定」というキーワードでくくり、メモリ環境のカスタマイズや周辺機器の接続など、初級から中級レベルまでの環境設定を解説していきます。その性質上どうしてもメモリの構成などについての理解が必要となっていますが、内容は非常にやさしくできあがっていますので恐れることはありません。とにかくパソコンの前に座って環境設定を始めてみましょう。

カバーデザイン
◆
太田公士(Ladybird)

Kobunsha

PC/AT互換機対応

CONFIG.SYSがわかる本

宮田　茂

もうムダなメモリは使わない

CONFIG.SYSとAUTOEXEC.BATの

修・正・加・工・自・由・自・在

●本書で使用したハードウェア

1.　IBM ThinkPad 230Cs
　　CPU　486/33MHz
　　メモリ　12M
　　PCMCIAMODEM：TDK DF1414
　　ETHERNET：TDK LAK-CD021
2.　自作 DOS/V 機
　　CPU　486/66MHz
　　メモリ　16M
　　SCSIADAPTEC 1522
　　CD-ROM 内蔵 SCSI
　　サウンドカード SoundBlaster 16 ASP
OS：MS-DOS Ver.6.2/V,PC-DOS Ver.6.3/V
　　Microsoft Windows3.1

●権利について

●内容について

本書は、著者ならびに編集部が上記の環境で実際に操作した結果を慎重に検討し、著述・編集したものです。本書の内容に関して、記載漏れ・誤り・疑問点・質問等がありましたら、出版元まで必ず書面（お手紙または FAX）にてご連絡下さい。また、ソフトの機能や操作については、直接メーカーにお問い合わせ下さい。

なお、本書の内容に関する運用結果については責任を負いかねますので、ご了承下さい。

●本文中の□について

本文で□の記号がついている箇所は、実際の入力では半角アキです。スペースキーを一回押して下さい。

まえがき

DOS/V パソコンは、ハードウェア的にはオープンなアーキテクチャで全世界的に普及しているパソコン IBM の PC/AT 互換機そのものです。つまり、英語圏のみならず、フランス語圏やドイツ語圏、東南アジアなど、世界中で使用されているパソコンの標準機を日本語で利用できるようになっているのです。

このようにいろいろな国のユーザーが、いろいろな言語を使用するという前提で設計されている PC/AT 互換機は、非常に高い汎用性を備えています。たとえば、使用する言語によって表示できる文字セットを簡単に変更することもできるし、日時の表示形式を指定したり、キーボードの配列を変更するといったカスタマイズも可能なのです。そのカスタマイズは、ハードウェアで対応するのではなくソフトウェアで対応していますから、特別なハードウェアを追加する必要はまったくありません。もちろん、ハードウェア的にみても、PC/AT のお作法を遵守している周辺機器なら自由に接続することができるのは言うまでもありません。

このように汎用性が高く、また拡張性に優れているということは、ユーザーサイドから見れば負担が高くなるだけですが、よく言われるように借金も財産のうちですから、この負担さえ支えきってしまえばその見返りとして輝くような自由度を得ることができるのです。

日本ではアルファベットを使用する言語圏と違い、漢字やひらがななど 2 バイトの文字を使用する関係で、この負担がさらに重くなっていますが、世界的な標準機を日本語で利用できるという魅力の前にはその負担など何者でもありません。

この日本語の特殊事情を鑑み、PC/AT 互換機用の特別な OS「DOS/

V」（MS-DOSの日本語バージョン）が開発されていますが、その特殊事情からどうしても設定が複雑になるという面は否定できません。日本語表示を行うためのソフトウェアを組み込まなければならないからです。

したがって、日本でDOS/Vパソコンを快適に利用していこうとするなら、日本語を利用するためのソフトウェアの設定や、周辺機器を利用するための設定、あるいはメモリのカスタマイズといった作業が必須となってきます。

本書では、これらすべてを「環境設定」というキーワードでくくり、メモリ環境のカスタマイズや周辺機器の接続など、初級から中級レベルまでの環境設定を解説していきます。その性質上どうしてもメモリの構成などについての理解が必要となっていますが、内容は非常にやさしくできあがっていますので恐れることはありません。とにかくパソコンの前に座って環境設定を始めてみましょう。

1995年8月　宮田　茂

第1部 CONFIG.SYSとAUTOEXEC.BATの基礎知識

第2部　CONFIG.SYSとAUTOEXEC.BATの基本コマンド

第3部　実例!周辺機器設定

第1部
CONFIG.SYSとAUTOEXEC.BATの基礎知識

環境構築を行ったりメモリをカスタマイズしたりするには、その基本的な知識とテクニックをマスターしておかなければなりません。余計なトラブルを防ぎ、もしも何かトラブルが発生した場合に冷静に対処するためです。また、最悪の場合、つまりパソコンがまともに起動できなくなった場合にそなえ、フロッピーディスクでパソコンを起動し、元の状態に復元できるような準備も必要です。さあ、まず準備から始めましょう。

第1章
DOS/V パソコンの環境設定に挑戦

「パソコンの環境を構築する」とか「MS-DOSをカスタマイズする」というような話をよく聞きます。でも実際、具体的にはどのようにすれば「環境を構築」できるのか「カスタマイズ」できるのか、よくご存知ないユーザーがほとんどです。いや、そもそも「そんな難しいことは・・・・・」とおっしゃる方も多いのではないでしょうか。

本書を一読すれば、DOS/Vパソコン（DOS/Vで動作するパソコン）の基本的な環境構築が行えるようになります。喰わず嫌いはやめて、環境構築に挑戦してみましょう。きっとますますご自分のパソコンがかわいく思えるようになるでしょう。
※コマンドの入力等の文書の中に、たとえば「DIR ▯/P ↵」の中にある▯はコマンドを入力する際に、半角アキ（スペースを1回押す）の印です。

Windows3.1かDOSか

Windows3.1はほぼOSに近い機能を持っていますが、まだまだMS-DOSの呪縛から逃れることができません。環境もMS-DOSの環境をほぼ引き継ぐこととなります。したがって、本書では基本的にMS-DOS上での環境構築が主体となりますので、もし、本書に沿ってパソコンをいじってみようというのでしたら、Windows3.1を終了しMS-DOSのプロンプトを表示させておいてください。その方法はご存知でしょうが、念のためその方法をご紹介しておきます。

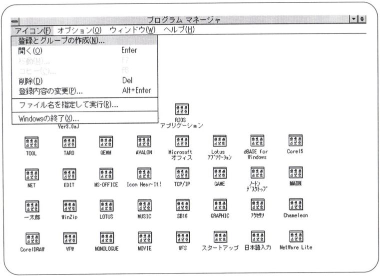
● MS-DOSのプロンプトを表示させる

Windows3.1のプログラムマネージャのコントロールボックスをダブル

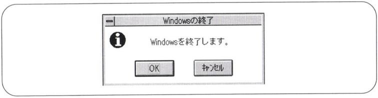
クリックするか、クリックして表示されるメニューから「閉じる（C）」を実行してください。または、「アイコン（F）」のメニュー（画面1）から、「Windowsの終了（X）」を実行してください。

「Windowsの終了」ダイアログボックスが表示されたら「OK」ボタンをクリックすればWindows3.1が終了し、MS-DOSのプロンプトが表示されます（画面2）。この状態でさまざまなDOSコマンドを実行することができます。

◎画面1　「アイコン（F）」のメニュー



◎画面2　「Windowsの終了」ダイアログボックス



起動ディスクを作成する

これからいろいろ環境を構築していくと、どうしてもパソコンが[1]**ハングアップ**しそのまま起動できないというようなことも起こってきます。その対策のために[2]**起動ディスク**を作成しておきましょう。

●起動ディスクを作成する

といっても起動ディスクの作成は非常に簡単です。新しいフロッピーディスクを用意し、ドライブ（普通はAドライブ）にセットしてから次のようにフォーマットするだけです。

```
FORMAT ␣ A:/S ↵
```

フォーマットが終わるといろいろとメッセージが表示されますが、重要なのは**「システムが転送されました」**というメッセージです。このメッセージが確認できれば、システムは転送されています（画面3）。

[1]パソコンが停止してしまい入力をまったく受け付けなくなる状態。
[2]パソコンを起動できるように MS-DOS のシステムを転送したフロッピーディスク。

◎画面３　システムが転送されたかどうかを確認する

```
C:¥>FORMAT A:/S
新しいディスケットをドライブ A: に入れてください
準備ができたら ENTERキーを押してください...

ディスクのフォーマット形式をチェックしています
アンフォーマット情報を保存しています
1.44M バイト　ベリファイ中です
フォーマットが終了しました
システムが転送されました

ボリューム・ラベルを指定してください
(半角11字まで,ラベルなしはENTERキー)

    1,457,664 バイト　：全ディスク空間
      145,920 バイト　：システム使用域
    1,311,744 バイト　：使用可能ディスク空間

          512:　クラスター当りのバイト数
        2,562:　ディスク中の使用可能クラスター数

ボリューム・シリアル番号は 3A2B-11DF です

別のディスクをフォーマットしますか (Y/N)?_

    英数 半角　R
```

「[3]**ボリュームラベル**を指定してください」と表示されますが、そのまま↵キーを押してしまってかまいません。

●起動できるか確認する

このディスクでは日本語環境もへったくれもありませんが、とにかくパソコンを起動するくらいはできます。念のためテストしてみてください。フォーマットしたディスクをフロッピードライブにセットし、パソコンを[4]**リセット**するとフロッピーディスクから起動されるはずです。

●ハードディスクから起動される場合

システムによっては、[5]**BIOS**のセットアップで[6]**ブートシーケンス**を指

[3]ディスク（ハードディスクでは１ドライブ）ごとにつけることのできる名前。
[4]リセットボタン、または Ctrl ＋ Alt ＋ Del でパソコンを再起動すること。
[5]「Basic In Out System」の略で、パソコンの基本的な設定をするためのプログラム。ROM に焼き付けられている。

定し、フロッピーディスクからの起動ができないようになっていることがありますので、AMIのBIOSを使用したシステムなら、BIOSセットアッププログラムの「ADVANCED SETUP」メニューで、「Floppy Drive Seek At Boot」を「Enabled」に、「System Boot Up Sequence」を「A：、C：」にセットし、フロッピーディスクで起動できるようにして下さい。

CONFIG.SYSとAUTOEXEC.BATってなんだぁ？

喰わず嫌いがまず食べなければならないのはたった2つのファイル、**「CONFIG.SYS」**と**「AUTOEXEC.BAT」**です。この「CONFIG.SYS」と「AUTOEXEC.BAT」という名前は誰でも一度ぐらいは聞いたことはあるでしょう。一冊のパソコン雑誌の中には必ずといっていいほど一回はこれらの名前が登場しているはずです。

それもそのはず、MS-DOSの環境はこの2つのファイルによって構築されるのです。

●環境構築のしくみ

パソコンの電源をONにすると、パソコンはハードディスク→フロッピーディスク、またはその逆の順でディスク上の特定の位置にアクセスします。そこにはIPL（Initial Program Loader）と呼ばれるプログラムが書き込まれていて、その指示によりOSを決定するのです。DOS/Vの場合はIPLが「このシステムはPC-DOS」であるとか「MS-DOSである」というように指示します。

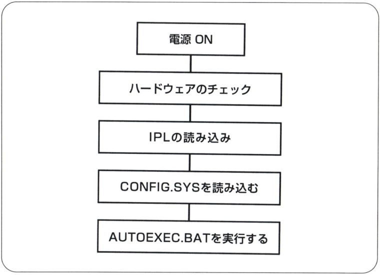
DOS/Vであるということが明確になると、次に「CONFIG.SYS」ファイルを読み込み、その内容にしたがってハードウェアとソフトウェアの環境を整えていきます。これでDOS/Vの起動は完了です。

続いて、AUTOEXEC.BATが読み込まれ、記述されている順番に一行づつ実行されます。つまり、AUTOEXEC.BATはDOS/Vが起動されてから初めて実行されるプログラムの手順書というわけです（図1）。

[6] どのドライブからパソコンを起動するか、その手順。

このように CONFIG.SYS で起動時の環境を整えますので、原則として CONFIG.SYS での設定は DOS/V 起動後に変更することはできませんが、AUTOEXEC.BAT はプログラムを実行する手順書に過ぎませんから、そこで設定された内容は起動後でもある程度自由に変更することができます。

◎図 1　MS-DOS 起動の手順



●存在を確認してみよう

DOS/V がインストールされているパソコンでは、必ず起動ドライブ（C ドライブ）の[7]**ルートディレクトリ**に「CONFIG.SYS」と「AUTOEX-EC.BAT」があります。確認してみてください。C ドライブのルートディレクトリを[8]**カレントディレクトリ**にして、「DIR ⏎」と入力してください(画面 4)。

ただし、ルートディレクトリにファイルやサブディレクトリがたくさん登録されていると、画面表示が一気に流れてしまい肝心の CONFIG.SYS と AUTOEXEC.BAT を確認することができませんので、その場合は

[7]一番上位のディレクトリ「C：¥」という状態。
[8]現在のディレクトリ。

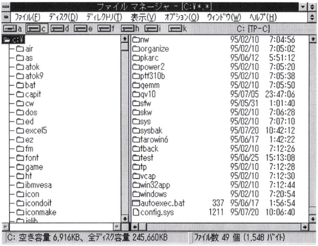
「DIR □/P ⏎」としてページごとに画面を停止して確認してください。もちろん Windows3.1 のファイルマネージャからも確認することはできます(画面５)。

◎画面４　ルートディレクトリにある

```
NW           <DIR>      95-02-10    7:04
ORGANIZE     <DIR>      95-02-10    7:05
PKARC        <DIR>      95-06-12    5:51
POWER2       <DIR>      95-02-10    7:05
PTF310B      <DIR>      95-02-10    7:05
QEMM         <DIR>      95-02-10    7:05
QV10         <DIR>      95-07-05   23:47
SFW          <DIR>      95-05-31    1:01
SKW          <DIR>      95-02-10    7:06
SYS          <DIR>      95-02-10    7:07
SYSBAK       <DIR>      95-07-20   10:42
TAROWIN6     <DIR>      95-06-17    1:42
TBACK        <DIR>      95-02-10    7:12
TEST         <DIR>      95-06-25   15:13
TP           <DIR>      95-02-10    7:12
VCAP         <DIR>      95-02-10    7:12
WIN32APP     <DIR>      95-02-10    7:12
WINDOWS      <DIR>      95-02-10    7:20
AUTOEXEC BAT          337 95-06-17    1:56
CONFIG   SYS        1,211 95-07-20   10:06
        49 個          1,548 バイトのファイルがあります
                  13,860,864 バイトが使用可能です

C:¥>_
   英数 半角  R
```

◎画面5 ファイルマネージャでも確認できる



●バックアップ

パソコンの環境を変更したり構築したりする前に、念のためこの CONFIG.SYS と AUTOEXEC.BAT の**複製**（バックアップ）を作成しておきましょう。もしも、なんかパソコンがおかしくなったり起動できなくなった場合は、ここで作成した複製が役にたちます。

バックアップは MS-DOS のコピーコマンドを使用します。「C:>」または「C:¥>」の状態（MS-DOS のプロンプト状態と言います）で次のように入力してください。

```
COPY □ CONFIG.SYS □ CONFIG.ORG ↵
```

続いて、AUTOEXEC.BAT も同じようにバックアップします。これもプロンプト状態で次のように入力してください。

```
COPY □ AUTOEXEC.BAT □ AUTOEXEC.ORG ↵
```

それぞれ「1個のファイルを複写しました」というメッセージが表示されれば大丈夫です（画面6）。

◎画面６　バックアップができている証拠

```
C:¥>COPY CONFIG.SYS CONFIG.ORG
        1 個のファイルを複写しました

C:¥>COPY AUTOEXEC.BAT AUTOEXEC.ORG
        1 個のファイルを複写しました

C:¥>_




    英数 半角   R
```

さらに念を入れて本当にバックアップが作成されたか見てみましょう。例によってプロンプト状態で「DIR ␣ CONFIG.＊↵」としてください。AUTOEXEC.BATの方も同様です。「DIR ␣ AUTOEXEC.＊↵」としてみてください。画面7のようになっていればいいでしょう。

◎画面７　バックアップされたファイルを確認する

```
C:¥>DIR CONFIG.*

 ドライブ C のボリュームは TP-C        です
 ボリューム・シリアル番号は 1ED6-0517 です
 ディレクトリーは  C:¥

CONFIG   SYS       1,211 95-07-20   10:06
CONFIG   ORG       1,211 95-07-20   10:06
        2 個             2,422 バイトのファイルがあります
                   6,762,496 バイトが使用可能です

C:¥>DIR AUTOEXEC.*

 ドライブ C のボリュームは TP-C        です
 ボリューム・シリアル番号は 1ED6-0517 です
 ディレクトリーは  C:¥

AUTOEXEC BAT         337 95-06-17    1:56
AUTOEXEC ORG         337 95-06-17    1:56
        2 個               674 バイトのファイルがあります
                   6,762,496 バイトが使用可能です

C:¥>_
    英数 半角   R
```

作成したバックアップ（拡張子がORGのもの）が見あたらない場合は、再度これまでの操作を繰り返してみてください。

●内容を確認する

CONFIG.SYSとAUTOEXEC.BATの存在を確認しバックアップを作成したら、その内容を見てみましょう。現在自分がどのような環境でパソコンを利用しているか知っておくことも大切です。

CONFIG.SYSとAUTOEXEC.BATは文字情報だけのテキストファイルですから、MS-DOSの**TYPEコマンド**でその内容を表示することができます。それぞれ次のように操作して内容を確認しましょう。

- **CONFIG.SYSの内容を表示する**

TYPE ␣ CONFIG.SYS ↵

- **AUTOEXEC.BATの内容を表示する**

TYPE ␣ AUTOEXEC.BAT ↵

もしも表示がスクロールしてしまう場合、今度は「/P」スイッチが使えませんので、MS-DOSの**MOREコマンド**を使います。「MORE ␣<

CONFIG.SYS↵」または「MORE □<AUTOEXEC.BAT↵」というようにします。

たとえば、IBM のサブノートパソコン Thinkpad230Cs にデフォルトで用意されている CONFIG.SYS と AUTOEXEC.BAT は画面 8、画面 9 のようになります。

◎画面 8　CONFIG.SYS の内容

```
BUFFERS=20
FILES=30
DOS=HIGH,UMB
COUNTRY=081,932,C:¥DOS¥COUNTRY.SYS
SHELL=C:¥DOS¥COMMAND.COM /P /E:512
DEVICE=C:¥DOS¥$FONT.SYS
DEVICE=C:¥DOS¥HIMEM.SYS
DEVICE=C:¥DOS¥EMM386.EXE RAM
DEVICEHIGH=C:¥DOS¥SETVER.EXE
DEVICEHIGH=C:¥DOS¥$DISP.SYS
DEVICEHIGH=C:¥DOS¥$IAS.SYS
DEVICE=C:¥EZGOING¥IBMDSS01.SYS /M
DEVICE=C:¥EZGOING¥IBMDOSCS.SYS
DEVICE=C:¥EZGOING¥DICRMU01.SYS
DEVICE=C:¥EZGOING¥$ICPMDOS.SYS
DEVICE=C:¥DOS¥POWER.EXE
DEVICE=C:¥EZGOING¥AUTODRV.SYS C:¥EZGOING¥AUTODRV.INI
DEVICE=C:¥EZGOING¥DISKDRV.SYS
DEVICEHIGH=C:¥DOS¥$PRN.SYS
DEVICEHIGH=C:¥DOS¥ANSI.SYS /X
DEVICEHIGH=C:¥DOS¥$FDD12.SYS
REM DEVICEHIGH=C:¥DOS¥RAMDRIVE.SYS
INSTALL=C:¥DOS¥IBMMKKV.EXE /M=S /Z=4 /C /L /J=90 /S=C:¥DOS¥MULTDICT.PRO /U=C:¥$U
-- つづく --
    英数 半角   R
```

◎画面９　AUTOEXEC.BAT の内容

```
C:¥>TYPE AUTOEXEC.BAT
@ECHO OFF
SET COMSPEC=C:¥DOS¥COMMAND.COM
REM - 以下の3行のコマンドは初期導入用に設定しておりますので編集時に
REM - 削除されても差し支えありません。
IF NOT EXIST C:¥F_IPL.COM GOTO SECOND
F_IPL
:SECOND
PROMPT $P$G
PATH C:¥WINDOWS;C:¥DOS;C:¥THINKPAD¥;C:¥IBMVESA;C:¥EZGOING;C:¥LLRA;%PATH%;
SET TEMP=C:¥DOS
SET IBMAV=C:¥DOS
SET SPEECHDATA=C:¥SPEECHW¥BIN
C:¥THINKPAD¥FUELDOS
C:¥DOS¥SMARTDRV.EXE
LH C:¥DOS¥KEYB.COM JP,932,C:¥DOS¥KEYBOARD.SYS
LH C:¥DOS¥PRNIBM.COM
REM C:¥DOS¥MOUSE.COM
CALL C:¥DOS¥IBMAVDR.BAT C:¥DOS¥
C:¥LLRA¥LLRA1.EXE
VER
WIN

C:¥>_
    英数 半角    R
```

もしも現在 Windows3.1 が起動しているなら、メモ帳を起動して CONFIG.SYS と AUTOEXEC.BAT を読み込んで確認できるはずです。

● CONFIG.SYS と AUTOEXEC.BAT が無いとどうなるか

パソコンの環境構築を語る場合、この「CONFIG.SYS」と「AUTOEXEC.BAT」は2つ一組で語られますが、とりあえずこの2つのファイルが存在しなくても MS-DOS でパソコンを起動することはできます。もっとも起動しても、化粧っけのないスッピンの MS-DOS ですので環境もなにもあったものではありませんからほとんど役にはたちません。

スッピンの DOS/V マシンでは日本語を表示することができませんしまったくの PC/AT 互換機＋MS-DOS（英語 DOS）ですし、せっかく搭載しているメモリを利用する設定も行われませんので、Windows3.1 はおろか MS-DOS のアプリケーションでも DOS5 以後のバージョンに対応しているものは動作しないものがほとんどです。

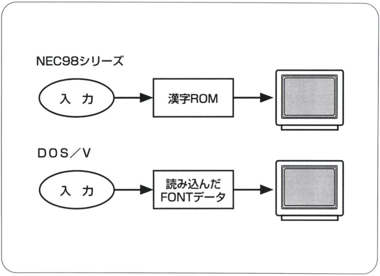
ただし、昔の8086時代の（16ビットの CPU の時代）英語アプリケーションを動かす、あるいは英語モードのゲームソフトなどでは逆にスッピンのままじゃないと動作しないものがあります。

● DOS/V の CONFIG.SYS

DOS/V は、MS-DOS（あるいは PC-DOS）という OS に日本語表示機能を付加したものですので、NEC の PC-98 シリーズで利用される MS-DOS と操作方法やコマンドの種類などほぼ同じ仕組みとなっています。

ところが、DOS/V は IBM の PC/AT 互換機で動作する MS-DOS（英語版）にソフトウェア的に日本語表示機能を付加したものですので、もともと漢字 ROM を登載していて、ハードウェア的に日本語に対応している PC-98 シリーズに比べると、日本語を表示するためのソフトウェアの設定が不可欠で、環境構築が複雑になるというデメリットがあります（図2）。

◎図２　DOS/V での日本語表示



その複雑な環境構築も CONFIG.SYS と AUTOEXEC.BAT の役目ですから、DOS/V の CONFIG.SYS と AUTOEXEC.BAT は PC-98 シリーズに比べ複雑怪奇と思われるのもしかたのないことです。また、CONFIG.SYS や AUTOEXEC.BAT が複雑になればなるほど、DOS/V の仕組みやメモリについてのより深い知識が必要となってきます。

第2章
環境構築基礎知識

パソコンショップの店頭で販売しているマシンは、さまざまなユーザーのニーズにマッチするように、最大公約数的な環境構築がなされています。最近では、ほとんどのDOS/VマシンにWindows3.1やそのアプリケーションがプリインストールされ、Windows3.1が動作する環境でセットアップされています。

このDOS/Vマシンを購入してきて、そのまま何の疑問も抱かずに使い続けることができるなら、特に環境を構築する必要もありません。が、実際は周辺機器を増設したり、メモリを増設したり、まったく購入した状態で使用することはほとんどないはずですので、どうしても多かれ少なかれ環境構築の世界に足を浸すことは必要となってくるのです。

ただし、あまり浸かりすぎると、環境設定のための環境設定を行う環境オタクになりかねません。ほどほどにしておきましょう。メモリを1バイト節約したからといって人生が変わるわけではありませんし、異性にもてるようになるわけでもありません。

メモリ構成の知識とMEMコマンド

では、まず環境構築をするにあたり、パソコンのメモリについて最低限の知識を仕入れておきましょう。

●基本は1Mバイト

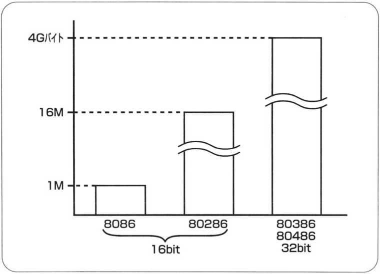
MS-DOSが管理できるメモリ領域は基本的にたった1Mバイトしかありません。これは、MS-DOSが16ビットCPU用のOSとして登場したため、現在でもその基本を引きずっているからです。

なぜ、16ビットの[9]**CPU** では1Mバイトまでしか管理できないかというと、当時の16ビット CPU（8086）には、メモリをアクセスするために必要なアドレス（メモリの番地）を指定するピンが20本しかなかったからでした。

つまり最大で**「2の20乗=1,048,576バイト=1Mバイト」**という計算となるのです。ちなみに32ビットのパソコンではアドレス用のピンも32本ありますから、**「2の32乗=約4Gバイト（4000Mバイト）」**という広大な空間を利用できるということになっています（図3）。

なお、同じ16ビット CPU でも後期の[10]**80286**はアドレス指定のためのピンを24本持っています。

◎図３　CPUのタイプによるアクセスできるメモリ空間の違い



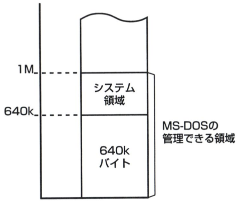
● 1Mバイト以上のメモリ（XMSとEMS）

32ビットのパソコンがいかに広大なメモリにアクセスできるといっても、使用するOSがMS-DOSでは直接管理できる領域は1Mバイト以下に限

[9]「Central Proccessing Cluit」の略　パソコンの頭脳のこと。
[10] 16bitのCPU。

られます。しかも、1Mの中の384KバイトはDOS/V自身が使用してしまいますので、純粋に利用できるメモリ（**コンベンショナルメモリ**と呼びます）は640Kバイトとなってしまいます（図4）。

◎図４　実際に利用できるのは640Kバイト



さらに、そのメモリ上にもDOS/Vを動作させるために必要なさまざまなソフトウェアが展開されますので、アプリケーションなどを動かすためのメモリは本当に少なくなってしまいます。

その状況を確認してみましょう。「MEM ⏎」とプロンプトの状態から入力してください（画面10）。ここに表示されている**「最大実行可能プログラムサイズ」**が、アプリケーションなどが使用できる最大メモリ容量です。

◎画面 10　MEM コマンドで確認する

```
C:¥>MEM

メモリーのタイプ    合計  =  使用済 +  使用可
----------------  -------   -------   -------
基本                640K      101K      539K
UMB                  63K       53K       10K
アダプタ RAM/ROM      0K        0K        0K
XMS* メモリー     10,164K    2,805K    7,359K
----------------  -------   -------   -------
システム全メモリ  10,867K    2,959K    7,908K

1MB 以下全メモリ    703K      154K      549K

全 EMS メモリー                     10,544K (10,797,056 バイト)
使用可能 EMS* メモリー               7,552K  (7,733,248 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ          539K (551,616 バイト)
使用可能最大 UMB メモリー               6K   (5,968 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

ではそれ以上（1M バイト以上）のメモリはどうなってしまうのか？　という疑問がふつふつとわいてきますが、ご心配はいりません。これら 1M バイト以上のメモリを**「プロテクトメモリ」**とか**「XMS メモリ」**と呼んでいますが、その XMS メモリを使えるようにするシステムがいろいろと考案されています。そのなかから環境構築にかかせない仕様をご紹介します。

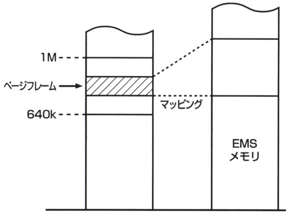
EMS（Expanded Memory Specification）

コンベンショナルメモリとして利用される 640K バイトの上から MS-DOS が管理できる 1M バイトまでの間に（物理アドレス A0000〜FFFF-Fh で「[11]**アッパーメモリエリア**」と呼ばれています）、連続した 64K バイトの領域（[12]**ページフレーム**）を設定し、その領域に広大なメモリの一部をマッピングする方法が EMS と呼ばれる仕様です（図5）。

---

[11]図7。
[12]図5参照。

◎図5　EMSのしくみ

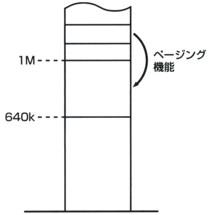


つまり、64Kバイトの窓にデータを投影してそのデータを眺めるということですが、この窓に投影するデータをどんどん切り替えていけば、ノートの隅に書いたパラパラアニメーションみたいに連続した領域として広大なメモリ空間をアクセスすることができるのです。

XMS (eXtended Memory Specification)

1Mバイト以上のメモリを管理する拡張メモリの規格です。基本的な仕組みはEMSと同じようなものですが、EMSでは実際のメモリ空間に窓をあけるのに対し、XMSの方は**ページング機能**を使います（図6）。

◎図6　XMSの仕組み



● UMBとHMA

さらに効率よくメモリを利用するためのキーワードとして、**UMB**と**HMA**という言葉があります。この2つのメモリ領域を有効に活用することで、コンベンショナルメモリにより余裕を持たすことができます。

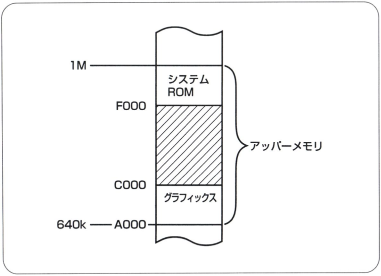
■ UMB（Upper Memory Block）

先ほどEMSのページフレームを設けた、640Kバイトから1Mバイトまでのアッパーメモリエリアはシステムが利用しますと言いました。このシステムというのは、文字やグラフィックを画面に表示するための**RAM**や、周辺機器を増設した際の**ROMアドレス**が割り当てられますが、384Kすべてが利用されているわけではありません。EMSのページフレームを設けることができたのもそのおかげですが、その空いている領域にXMSメモリを割り当てて利用してしまおうというのが「**UMB**」です。

何も周辺機器を増設していない状態では、このUMBは物理アドレスC0000~EFFFFhまでの192Kバイトです（図7）。ただし、システムによってはHMAにUMBを割り当てることもできるので、192Kバイトが

最大というわけではありません。

◎図７　アッパーメモリエリア



もちろん、この192Kバイトをそっくりそのまま使えるはずはなく、たとえばEMSを設定していればそのページフレームに64Kバイトとられます。

UMBの利用状況はやはり**MEMコマンド**で見ることができます。「MEM/D/P↵」と入力してみましょう（画面11）。

◎画面 11　UMB の利用状況

```
  03053           457,408  (447K)  IBMDOS       -- 空き --

UMB メモリーの詳細：

  アドレス  領域     サイズ          名前         タイプ
  -------  ------  ---------------- ------------ --------
  0C149       1     54,048   (53K)  IBMBIO       システム・データ
                    26,256   (26K)    $IBMADSP   デバイスドライバー $DISP
                     6,896    (7K)    $IBMAIAS   デバイスドライバー $IAS
                     1,056    (1K)    $IBMAPRN   デバイスドライバー $PRN
                     4,464    (4K)    CON        デバイスドライバー ANSI
                    15,296   (15K)    $IBMAF12   デバイスドライバー $FDD12
  0CE7C       1      6,176    (6K)  IBMDOS       -- 空き --

  0E901       2      4,080    (4K)  IBMDOS       -- 空き --

メモリーの概要：

  メモリーのタイプ    サイズ   =   使用済   +   使用可能
  ----------------  ----------  ----------  ----------
  基本                655,360     103,568     551,792
  UMB                  64,320      54,064      10,256
  アダプタ RAM/ROM          0           0           0
準備ができたらキーをどれか押してください . . ._
    英数 半角   R
```

HMA

この HMA はコウモリみたいなメモリで、実際は XMS の領域に属しますが 1M バイト以下の DOS が管理できる領域と同じような使い方ができるメモリです。これは、MS-DOS のメモリアドレス指定方法のマジックです。

8086ではアドレス用のピンが20本あるにもかかわらず、アドレスを指定できるのは16ビットですから、20ビット分のアドレスを表すために上位16ビット（セグメント）と下位16ビットに分けて指定します。

たとえば、「CFFF:FFFF」というように表現します。これを実際に物理アドレスに換算するわけですが、その方法は1バイトだけずらして足し算を行います。

```
 CFFF    ←セグメント
+ FFFF   ←オフセット
--------       DFFEF   ←実アドレス
```

ところがこの方法では最大「FFFF:FFFF」までアクセスできることに

なります。これを10進数になおすと「1,114,095」となり、20ビットで表現できる最大アドレス「FFFFF（1,048,575)」よりも65,520バイト（約64K）多くなってしまいます。

この64Kバイトは、実際はアクセスできる領域でありならがなんとなくアクセスできないようになっていたというコウモリメモリなのです。

実際は使用できるメモリですから、ここにMS-DOSのシステムをおいておくことが一般的です。これも「MEM⏎」で見てみましょう（画面12)。「DOSはHMA領域に常駐しています」というのが確認のポイントです。

◎画面12　DOSがHMAに常駐している

```
C:¥>MEM

メモリーのタイプ    合計 = 使用済 + 使用可
----------------  -------  -------  -------
基本                640K     101K     539K
UMB                  63K      53K      10K
アダプタ RAM/ROM      0K       0K       0K
XMS* メモリー      10,164K   2,805K   7,359K
----------------  -------  -------  -------
システム全メモリ  10,867K   2,959K   7,908K

1MB 以下全メモリ    703K     154K     549K

全 EMS メモリー                 10,544K (10,797,056 バイト)
使用可能 EMS* メモリー           7,552K  (7,733,248 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ        539K (551,616 バイト)
使用可能最大 UMB メモリー            6K   (6,176 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

環境構築の必要性

では、どういった場合に環境構築が必要となってくるのでしょうか。

●メモリを増設した場合

メモリを増設すると、その増設したメモリ分をどのように利用するか、あるいはこれまでのメモリ使用方法の配分を変更するかが必要なことがあります。

たとえば、EMSメモリをどれくらいとるのか、あるいは[13]**SMARTDRV**にどれくらいのメモリを割り当てるのかなどなど、搭載しているメモリの全体量に対して最適な配分方法を考え環境構築を行わなければなりません。

●周辺機器を増設・変更する場合

周辺機器を増設した場合、必ずといっていいほどその周辺機器を動作させるために環境を構築し直さなければなりません。[14]**ビデオカード**などの基本的な機器を変更するくらいでしたら、特に環境構築を行わなくていい場合もありますが、[15]**SCSIカード**でさえも特別な**ドライバソフト**（ハードウェアを制御するソフトウェア）をCONFIG.SYSで組み込まなければなりません。

また、ノートパソコンでは標準となっている[16]**PCMCIAカード**を利用する場合は、まずそのカードを利用できるようなドライバを組み込み、さらにハードウェア特有のドライバをインストールしなければなりません。

将来的には周辺機器を増設しても、OS（Windows95など）が自動的にそれをサーチして自動的に環境を構築してくれるようになりますが、現状ではまだまだ自分で環境を構築しなければならないようになっています。

---

[13]「ディスクキャッシュプログラムを組み込む」を参照。
[14]画面表示のための周辺機器。
[15]SCSI機器用のインタフェイス。
[16]「PCMCIAカードを使う」を参照。

●ソフトウェアをインストールした場合

ソフトウェアによっては特別な環境を必要とするものがあります。たとえば、ワープロソフトの一太郎を利用するならかな漢字変換機能のATOKが欠かせませんし、メモリ関係のカスタマイズも必要です。

もちろんソフトウェアをインストールする際に、自動的にある程度の環境は構築してくれますが、ユーザーの思い通りの環境に設定されるとは限りませんので、やはり環境設定を検証する必要が出てきます。

環境ってなんだぁ？

ここまで気安く「環境」などという言葉をつかってきましたが、それでは「環境とは何か」と尋ねられると少々困ってしまいます。

パソコンのメモリ容量、ハードディスク容量、増設している周辺機器など千差万別です。このようなハードウェアを含めたシステム全体を広い意味でパソコンの「環境」と呼びます。

また、これらを快適に利用するためには、メモリの利用方法を指定したり、周辺機器を使うための設定を行ったりしなければなりませんが、その設定を「環境設定」という言葉で表しています。

まだまだ霧の中にいるあなたのために、具体的な環境設定について解説しましょう。環境設定を大きく分けると次のようなジャンルがあります。

●基本的な環境設定（OSが必要とする環境設定）

OSをインストールすると自動的にCONFIG.SYSとAUTOEXEC.BATが作成されます。そのCONFIG.SYSとAUTOEXEC.BATの中でも、OSが快適に動作するために必要なソフトウェアなどが基本的な環境にあたります。

もちろん、DOS/VはCONFIG.SYSとAUTOEXEC.BATがなくても起動することはできますが、とてもじゃないがDOS/Vとは呼べないようなものです。

基本的な環境は、CONFIG.SYSの**BUFFERS**や**FILES**、**SHELL**コマンドなどで設定するものです。

●メモリ環境設定

DOS/V マシンに限らず、最近のパソコンは Windows を意識して最初から多量のメモリを搭載しています。さらに Windows 上で快適に作業を行うには、12M バイトや 16M バイトといったメモリ量が必要ですので、メモリ増設を行っているユーザーも多いことでしょう。

メモリを増設すれば、それだけメモリの用途が広がりますので、環境を構築する選択肢も多くなります。

●日本語表示の環境設定

何度も言うようですが、DOS/V ではソフトウェア的に日本語表示機能を付加していますので、そのための設定を行わなければなりません。この設定が不的確だと、日本語を表示することができなくなります。

●周辺機器の環境設定

周辺機器を増設する場合、その周辺機器が適正に動作するような設定を行わなければなりません。身近なところでは、SCSI カードの増設、**CD-ROM ドライブ**や**サウンドカード**の増設などがこれにあたります。

●日本語入力環境の設定

かな漢字変換機能は、通常 CONFIG.SYS にそのドライバを記述して組み込みます。

●ソフトウェアの実行環境設定

ソフトウェア独自の環境が必要なこともありますが、通常 AUTOEXEC.BAT でその環境を設定することができるので、後から直接コマンドや[17]**バッチファイル**によって変更・設定することも可能です。

[17]処理の手順を記述したテキストファイル。

コマンドとデバイスドライバ

以上のような環境をCONFIG.SYSとAUTOEXEC.BATで構築していくわけですが、CONFIG.SYSではCONFIG.SYS特有のコマンドとデバイスドライバの組み込みを行います。

AUTOEXEC.BATは、起動時に自動実行されるバッチファイル（操作の手順がかかれたもの）ですので、コマンドのみで構成されます。

●コマンド

MS-DOSのコマンド、AUTOEXEC.BATで使用されるコマンドは原則的にキーボードから入力することができますが、CONFIG.SYS特有のコマンドはキーボードから入力することはおろか、MS-DOSが起動してしまった後では実行することはまったくできません。

●デバイスドライバ

XMSメモリを利用できるようにするデバイスドライバ「**HIMEM.SYS**」やEMSのドライバ「**EMM386.EXE**」、日本語入力環境を利用するためのドライバ、さらにはCD-ROMなどのハードウェアを利用するためのドライバもCONFIG.SYSで指定しなければなりません。

なお、デバイスドライバには「**キャラクタ型**」と「**ブロック型**」という2つのタイプがあって、キャラクタ型は**ADDDRV**・**DELDRV**コマンドで自由に組み込んだり取り外したりすることができます（MS-DOS）。

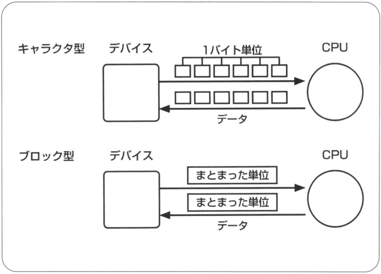
■ キャラクタ型

1バイト単位で入出力を行うようなデバイスドライバのことです。たとえば、かな漢字変換機能のデバイスドライバなどがこれにあたります。

■ ブロック型

ある一定の単位でアクセスが行われるデバイスドライバです。RAMディスクなどディスク関連のデバイスドライバはブロック型です（図8）。

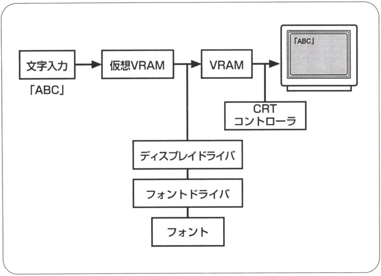
◎図８　キャラクタ型とブロック型デバイス



画面への日本語表示のしくみ

DOS/V ではソフトウェア的に日本語表示機能を付加しています。NEC の PC-98 シリーズなどのもともと日本語仕様のパソコンでは、本体に日本語のフォントを ROM として持ち、表示したい文字のパターンをその ROM から読みだしてテキスト画面に表示しますが、DOS/V ではフォントファイルからあらかじめフォントを RAM に読み込んでおき、そこから文字パターンを生成してグラフィック画面に表示します（図９）。

◎図9　DOS/Vの画面表示



このように、DOS/Vではソフトウェア的に日本語を表示しているため、画面での日本語表示速度が若干落ちるデメリットがありますが、現在のハードウェア性能をもってすればそう気になるほどではありません。

また、Windows環境となれば、テキスト画面もグラフィック画面もありませんから、そのデメリットはほとんど関係ないと思ってもいいでしょう。

第3章
実践基本設定講座

知識は実践されて初めて意味を持ちます。知識もずいぶんと仕込みましたし、さあ実際に環境設定を行ってみましょう。そのまえに基本テクニックです。

起動できなくなった時の処理

これからCONFIG.SYSとAUTOEXEC.BATにさまざまな編集や修正を加えていきますが、もしもそれが原因でパソコンのシステムが起動できなくなった場合、次の3つの方法のどれかでとにかくパソコンを起動してください。

●起動するための3つの方法

■ 起動ディスクで起動する

最初に作成した起動ディスクでシステムを起動し、バックアップしたCONFIG.SYSとAUTOEXEC.BATを元に戻して起動するという手段が一番確実です。

■ 選択しながら起動

パソコンの起動時、最初に表示される「Starting　PC-DOS」または「Starting MS-DOS」のところで **F8** キーを押してください。CONFIG.SYSとAUTOEXEC.BATを一行ずつ選択して実行することができます(画面13)。

◎画面１３　一行ずつ選択して実行する

```
Starting PC DOS...

PC DOS will prompt you to confirm each CONFIG.SYS command.
DOS=HIGH,UMB [Y,N]?Y
BUFFERS=20 [Y,N]?Y
FILES=30 [Y,N]?Y
COUNTRY=081,932,C:¥DOS¥COUNTRY.SYS [Y,N]?Y
SHELL=C:¥DOS¥COMMAND.COM /P C:¥DOS [Y,N]?Y
DEVICE=C:¥DOS¥HIMEM.SYS [Y,N]?Y
DEVICE=C:¥DOS¥EMM386.EXE RAM [Y,N]?Y
DEVICEHIGH=C:¥DOS¥$FONT.SYS /P=C:¥FONT [Y,N]?Y
Loading font files ...
$FONT.SYS Version 1.21 was installed.
DEVICEHIGH=C:¥DOS¥$DISP.SYS  [Y,N]?Y
$DISP.SYS Version 1.32 が導入されました。
DEVICEHIGH=C:¥DOS¥$IAS.SYS [Y,N]?
```

これはCONFIG.SYSとAUTOEXEC.BATの間違いを捜すのに便利です。どの行が原因でマシンの起動に失敗したのか、ある程度目鼻をつけることくらいはできますが、手間はかかります。

■ CONFIG.SYSとAUTOEXEC.BATをスキップする

[F8]キーのかわりに[F5]キーを押すと、CONFIG.SYSとAUTOEXEC.BATをスキップして起動することができますが、メモリ設定などによってはハングアップすることもあります。それでも一度は試す価値はあるでしょう。

● CONFIG.SYSとAUTOEXEC.BATを書き戻す

マシンが起動したら、次のように操作し元のCONFIG.SYSとAUTOEXEC.BATに戻してください。

```
COPY □ CONFIG.ORG □ CONFIG.SYS ↵
COPY □ AUTOEXEC.ORG □ AUTOEXEC.BAT ↵
```

この操作を行ったら、TYPEコマンドでCONFIG.SYSとAUTOEX-

EC.BAT の中身を確認してからパソコンをリセットしてください。正常に起動できるはずです。

もしも、起動できなかった CONFIG.SYS と AUTOEXEC.BAT を保存して後の参考にするなら、前記の操作を行う前に次のようにして現在の CONFIG.SYS と AUTOEXEC.BAT を保存してください。

```
COPY ␣ CONFIG.SYS ␣ CONFIG.OLD↵
COPY ␣ AUTOEXEC.BAT ␣ AUTOEXEC.OLD↵
```

MEM コマンドを使いこなす

実践で必要となる情報のうち、最も重要なのはメモリの使用状況です。その情報を得るためには、DOS/V の MEM コマンドを使いこなさなければなりません。これまでも MEM コマンドはたびたび登場し断片的にその機能を発揮してきましたが、MEM コマンドは重要なツールですからここでまとめてその使い方を復習しておきましょう。

● MEM を使ってみる

とにかく「MEM↵」とやってみましょう。画面14のようにメモリ使用状況の概要が表示されます。

◎画面１４　MEM コマンドでメモリ状況を見る

```
C:¥>MEM

メモリーのタイプ      合計  =  使用済 +  使用可
----------------   -------   -------   -------
基本                  640K      101K      539K
UMB                    63K       53K       10K
アダプタ RAM/ROM        0K        0K        0K
XMS* メモリー       10,164K    2,805K    7,359K
----------------   -------   -------   -------
システム全メモリ    10,867K    2,959K    7,908K

1MB 以下全メモリ       703K      154K      549K

全 EMS メモリー                        10,544K (10,797,056 バイト)
使用可能 EMS* メモリー                  7,552K  (7,733,248 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ              539K (551,616 バイト)
使用可能最大 UMB メモリー                  6K   (5,968 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

基本

コンベンショナルメモリ（640K バイト）のことです。**「ユーザーメモリエリア」**などと呼ばれることもあります。

UMB

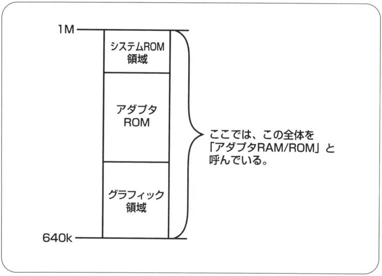
640K バイトから 1M バイトまでのメモリ領域ののうち、周辺機器が使用する分を除いたメモリ領域です。

アダプタ RAM／ROM

通常は、640K バイトから 1M バイトまでのメモリ領域のうち、システム ROM 領域と画面表示用のメモリ領域を除いた分のメモリを**「アダプタ ROM 領域」**と呼んでいます。この領域のうち、空いているところは UMB として使用しますので、当然「使用可」のメモリはゼロとなっています。

ただし、ここで表示されている**「アダプタ RAM/ROM」**はいわゆるアッパーメモリ全体のことを指し示しています（図10）。

◎図10　アダプタ RAM/ROM



XMS メモリ

1M バイト以上のメモリ搭載量です。このメモリ量と**「使用可能 EMS メモリー」**の値はシステムが適当に判断して割り振りを変更します。

全 EMS メモリー

EMS メモリの総量です。

使用可能 EMS＊メモリー

全 EMS メモリから、現在すでに使用している EMS メモリをのぞいたメモリ量が示されていますが、この量は可変です。

実行可能プログラム最大サイズ

現在使用できるコンベンショナルメモリの総量です。いくら EMS メモリや XMS メモリに余裕があっても、このコンベンショナルメモリが少ないと実行できないプログラムが出てきます。

使用可能最大 UMB メモリー

現在使用できる UMB メモリの総量です。

● MEM コマンドのオプション

MEM コマンドでは、現在のメモリ使用状況の概要を示すことができますが、さらに詳しい状況を知りたい場合はオプションを使います。

MEM コマンドのオプションは、「MEM □/?⏎」で表示させることができます（画面15）。

◎画面 15　MEM コマンドのオプション

```
C:¥>MEM /?
システム内の使用/空きメモリー量を表示します。

MEM [/CLASSIFY | /DEBUG | /FREE | /MODULE ﾌﾟﾛｸﾞﾗﾑ名] [/PAGE]

  /CLASSIFY (/C)   メモリー使用方法によってプログラムを分類します。
                   プログラム・サイズ，使用中のメモリーの概略，使用可能最大
                   メモリー・ブロックのリストを一覧表示します。
  /DEBUG (/D)      プログラム，内部ドライバーや，その他の情報を表示します。
  /FREE or /F      基本メモリーとUMBメモリーについて使用可能なメモリーの
                   サイズに関する情報を表示します。
  /MODULE (/M)     指定したプログラムが使用するメモリーの詳細を表示します。
                   このオプションの直後に表示したいプログラム名を指定します。
  /PAGE (/P)       画面単位で一時停止します。

C:¥>_



     英数 半角   R
```

■ /C(/CLASSIFY)（画面16）

◎画面16 /C

```
1MB 以下のメモリーを使用しているプログラム／ドライバー：

  名前          合計       =     基本メモリー    +    UMB メモリー
  --------  ----------------   ----------------   ----------------
  SYSTEM      18,093   (18K)     17,997   (18K)        96    (0K)
  $FONT        4,880    (5K)      4,880    (5K)         0    (0K)
  HIMEM        1,072    (1K)      1,072    (1K)         0    (0K)
  EMM386       4,112    (4K)      4,112    (4K)         0    (0K)
  POWER        4,560    (4K)      4,560    (4K)         0    (0K)
  IBMMKKV     22,288   (22K)     22,288   (22K)         0    (0K)
  COMMAND      4,496    (4K)      4,496    (4K)         0    (0K)
  VGACAP       5,056    (5K)      5,056    (5K)         0    (0K)
  SMARTDRV    29,360   (29K)     29,360   (29K)         0    (0K)
  KEYB         6,496    (6K)      6,496    (6K)         0    (0K)
  PRNIBM       3,232    (3K)      3,232    (3K)         0    (0K)
  $DISP       26,256   (26K)          0    (0K)    26,256   (26K)
  $IAS         6,896    (7K)          0    (0K)     6,896    (7K)
  $PRN         1,056    (1K)          0    (0K)     1,056    (1K)
  ANSI         4,464    (4K)          0    (0K)     4,464    (4K)
  $FDD12      15,296   (15K)          0    (0K)    15,296   (15K)
  - 空き -   562,048  (549K)    551,792  (539K)    10,256   (10K)

準備ができたらキーをどれか押してください . . ._
    英数 半角   R
```

現在メモリ上にロードされているプログラムが、どのようなメモリをどれだけ使用しているのかを知ることができます。

/D(/DEBUG)（画面17）

◎画面 17　/D

```
基本メモリーの詳細：

アドレス        サイズ          名前       タイプ
-------    ---------------  ----------  --------
00000       1,023   (1K)               割込みベクトル
00040         255   (0K)               ROM 通信域
00050         511   (0K)               DOS 通信域
00070       3,232   (3K)   IBMBIO      システム・データ
                            CON        システム・デバイスドライバー
                            AUX        システム・デバイスドライバー
                            PRN        システム・デバイスドライバー
                            CLOCK$     システム・デバイスドライバー
                            A: - D:    システム・デバイスドライバー
                            COM1       システム・デバイスドライバー
                            LPT1       システム・デバイスドライバー
                            LPT2       システム・デバイスドライバー
                            LPT3       システム・デバイスドライバー
                            COM2       システム・デバイスドライバー
                            COM3       システム・デバイスドライバー
                            COM4       システム・デバイスドライバー
 0013B      5,152   (5K)   IBMDOS      システム・データ
 0027E     22,320  (22K)   IBMBIO      システム・データ
準備ができたらキーをどれか押してください . . ._
   英数 半角   R
```

現在のメモリ状況をアドレス順に並べて表示することができます。なお、このアドレスは[18]**オフセット**値ですから実アドレスにするには10倍（16進で）しなければなりません。

[18]アドレス表記の上位5ケタ。

/F(/FREE)（画面18）

◎画面 18　/F

```
C:¥>MEM /F

使用可能基本メモリー：

  アドレス      サイズ
  -------   -----------------
   007F7          80    (0K)
   00E65          80    (0K)
   01952         208    (0K)
   0195F      94,016   (92K)
   03053     457,408  (447K)

使用可能合計: 551,792  (539K)

使用可能 UMB メモリー：

  領域    最大使用可能      使用可能合計       全サイズ
  ------  --------------  --------------  --------------
     1     5,968   (6K)    6,176   (6K)   60,240  (59K)
     2     4,080   (4K)    4,080   (4K)    4,080   (4K)

C:¥>_

   英数 半角  R
```

現在のコンベンショナルメモリと UMB メモリで利用可能な領域とサイズを表示します。

/M プログラム名（/MODULE）

メモリ上に常駐しているプログラムが実際どのアドレスからどれだけの領域を使用しているかを表示することができます。たとえば、COMMAND.COM について調べるなら、「MEM □/M □ COMMAND↵」とします（画面19）。

◎画面１９　/M で COMMAND.COM の常駐状況を表示する

```
C:¥>MEM /M COMMAND

COMMAND は以下のようにメモリーを使用しています：

  アドレス  領域      サイズ          タイプ
  -------  ------  ---------------  --------
   00D6D            3,968   (4K)   プログラム
   00E6A              528   (1K)   環境
                   ---------------
  合計サイズ：       4,496   (4K)

C:¥>_



    英数 半角   R
```

なお、プログラム名には[19]**拡張子**をつけてはいけません。拡張子をつけると画面20のようなエラーメッセージが表示されます。

[19]MS-DOS ファイル名は「××××××××.YYY」という形式ですがピリオドの後ろ「YYY」を拡張子と呼びます。

◎画面 20　エラーメッセージ

```
C:¥>MEM /M COMMAND

COMMAND は以下のようにメモリーを使用しています：

  アドレス  領域        サイズ          タイプ
  -------  ------  ----------------  --------
  00D6D             3,968    (4K)  プログラム
  00E6A               528    (1K)  環境
                   ----------------
  合計サイズ：      4,496    (4K)

C:¥>MEM /M COMMAND.COM
COMMAND.COM は現在メモリー上にロードされていません。
                ↑
C:¥>_      COMが余計な場合のエラーメッセージ

    英数 半角   R
```

/P(/PAGE)

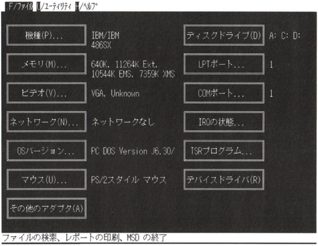
このオプションはもうすでに何度も登場しているのでもうご存じですね。MEM コマンドで表示させたい情報がスクロールしてしまうとき、このオプションをつければ画面単位で停止させることができます。

「準備ができたらキーをどれか押してください」と表示されているように、停止した画面を再度スクロールさせるためには何かキーを押しますが、このような場合通常↵キーが一番安全です。

MSD コマンドを使う

MSD コマンドは、メモリ専門のツールではありませんが、パソコンの現在の状況を広く知ることができる非常に便利なツールです。起動は「MSD ⏎」とするだけです（画面21）。

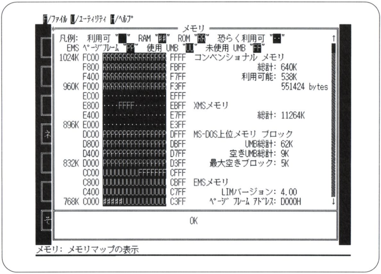
◎画面 21　MSD コマンドの起動画面



●メモリ状況の表示

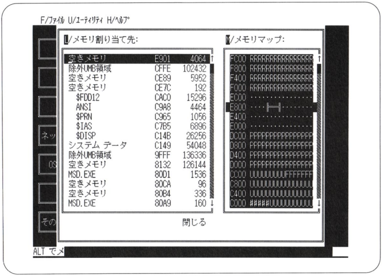
この画面で［M］キーを押せば現在のメモリ状況を見ることができます（画面22）。

◎画面 22　現在のメモリ状況を見る

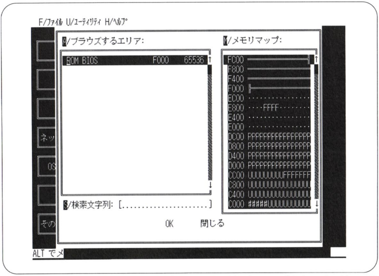


さらに詳しい情報を知りたい場合は、**Alt**＋**U** キーでユーティリティのメニューを表示させ、「M/メモリブロック表示」（画面23）または「B/メモリブラウザ」（画面24）を選択してください。

◎画面23　メモリブロックを表示させる



◎画面24　メモリブラウザ



メモリブロックの表示では、現在のメモリ状況をマップとして表示し、どのプログラムがどの領域を使用しているかが一目瞭然となるような仕組みで

すから、メモリ状況が一層よく把握できるようになっています。適時利用してみてください。

ADDDRV/DELDRVを使いこなす（MS-DOS）

MS-DOSには**ADDDRV**コマンドが附属しています。このADDDRVでは、マシンを起動後に[20]**デバイスドライバ**を組み込むことができるのです。ただし、組み込めるデバイスドライバはキャラクタ型のデバイスドライバのみです。

起動時にCONFIG.SYSで組み込んでおけば、わざわざ後から組み込む必要はないのですが、たとえば、Windowsを使用する環境ではDOS版のかな漢字変換を外しておいて、後で必要なときにADDDRVで組み込むといった用途に使用します。もちろん、その組み込み方も簡単ですので、日頃不要なドライバなどを必要に応じて組み込むようにすると、メモリの有効活用をはかることができます。

●定義ファイルを作成する

ADDDDRVは、特別に作成されたファイルの内容を読み出し、その内容にしたがって環境を構築していきますので、まず**定義ファイル**を作成しなければなりません。

この定義ファイルは、実はCONFIG.SYSでデバイスドライバを組み込む場合とまったく同じ形式、同じ書式なのです。たとえば、ATOK8を組み込む定義ファイルでしたら次のようになります。[21]**エディタ**やワープロで作成してください。作成したら、このファイルに適当な名前をつけて保存しておきましょう。ここではATOK.DEVとしておきます。

[20]周辺機器を適正に動作させるためのソフトウェア。
[21]ワープロのように文字を入力するツールですが、文字飾りなどの装飾機能はなく、もっぱら高速に文字を入力することに使われます。もともとはプログラムを書くツールでした。

```
DEVICEHIGH = C:¥ATOK8¥ATOK8A.SYS↵
/UCF = C:¥ATOK8¥ATOK8.UCF↵
DEVICEHIGH = C:¥ATOK8¥ATOK8B.SYS↵
DEVICEHIGH = C:¥ATOK8¥ATOK8IAE.SYS/P = 3↵
```

●デバイスドライバを組み込む

ADDDRV で組み込めるのは「一つだけ」ですが、この一つは「一つのデバイスドライバ」ということではなく、「一つの定義ファイル」という意味です。したがって、この例のように、一つの定義ファイルの中に複数のデバイスドライバを組み込む記述があれば、それらを一括して組み込むことができます。

組み込み方は簡単です。コマンドに続けて組み込む定義ファイルを指示すればいいのです。たとえば、先ほど作成した ATOK.DEV が C:¥ATOK というディレクトリに入っているなら、次のように実行します。

```
ADDDRV □ C:¥ATOK¥ATOK.DEV↵
```

なお、定義ファイルはプログラムではなくデータファイルですから、**パス**が通っているディレクトリにあるとしても、カレントディレクトリ以外では必ずフルパスで指定しなければなりません。

もしも、**パス**名を指定するのが面倒な場合は、バッチファイルを作成するか **APPEND コマンド**を使用します。

●バッチファイルの場合

バッチファイルは次のようになるでしょう。

```
@ECHO □ OFF↵
ADDDRV □ C:ATOK¥ATOK.DEV↵
```

もちろん、このバッチファイルもパスの通ったディレクトリでなければ何のためにバッチファイルを作成したのかわかりませんので、ちゃんとパスの通ったディレクトリに入れてください。

たとえば、このバッチファイルが ATOK.BAT というような名前でパス

の通っているディレクトリにあれば、「ATOK⏎」とするだけでATOK8を組み込むことができます。

● APPENDを利用する

環境構築において、**PATHコマンド**は非常に脚光を浴びているのに、APPENDはあまりなじみのないコマンドです。機能としてはPATHを上回るものがあり、案外便利な使い方もあるので覚えておくといいでしょう。形式は、ファイルなどを検索したいディレクトリ名を指定するだけです。たとえば、C:¥DEVというディレクトリにATOKの定義ファイルATOK.DEVがあるなら、次のように指定します。

```
APPEND □ C:¥DEV⏎
```

これでAPPENDされたディレクトリが環境変数APPENDにセットされます。確認は何もオプションをつけずに単に「APPEND⏎」と入力します（画面25）。

◎画面25　APPENDでディレクトリを指定した

```
C:¥>APPEND

C:¥>APPEND C:¥DEV

C:¥>APPEND
APPEND=C:¥DEV

C:¥>_
```

オープンしたいファイルがあるディレクトリにAPPENDを指定しておけば、どのディレクトリからでもADDDRVコマンドでATOKの定義フ

ァイルをオープンすることができます。

APPEND には PATH コマンドと同等の機能を実現するオプションなどもありますから、うまく使えば非常に便利です（画面26）。

◎画面 26　APPEND のオプションを見てみる

```
C:¥>APPEND /?
指定したディレクトリのデータファイルを現在のディレクトリにあるファイルと同じ
ように開けるようにします.

APPEND [[ﾄﾞﾗｲﾌﾞ:]ﾊﾟｽ[;...]] [/X[:ON | :OFF]] [/PATH:ON | /PATH:OFF] [/E]
APPEND ;

  [ﾄﾞﾗｲﾌﾞ:]ﾊﾟｽ  アペンドするドライブやディレクトリを指定します.
  /X:ON         アペンドされたディレクトリをファイルの検索やアプリケーションの
                実行ができるようにします.
  /X:OFF        アペンドされたディレクトリのファイルを開くことだけできるように
                します. /X:OFF は既定値です.
  /PATH:ON      パスが検索するファイル名に含まれているときもアペンドされたディ
                レクトリを検索するように指定します.  /PATH:ON が既定値です.
  /PATH:OFF     /PATH:ON の効果を無効にします.
  /E            アペンドされたディレクトリの一覧のコピーを環境変数 APPEND に
                記憶します. /E スイッチは, システムを起動してから最初に
                APPEND を使うときだけに使います.

パラメータとして ; (セミコロン)だけを指定すると, 現在設定されている検索パスを
すべて取り消します.
パラメータの指定がなければ, アペンドされているディレクトリの一覧を表示します.

C:¥>_
```

●取り外す

ADDDRV で組み込まれたデバイスドライバを取り外すには、**DELDRV コマンド**を実行します。こちらはオプションをつける必要はありません。

```
DELDRV↵
```

これだけでいいのです。ただし、ADDDRV でデバイスドライバを組み込んだ後で[22]**常駐型のプログラム**などを実行すると、DELDRV でデバイスドライバを取り外すことができなくなることがあります。

[22]実行するとメモリに常駐し、常に実行できるようなプログラム。「TSR」とも言います。

第2部
CONFIG.SYSとAUTOEXEC.BATの基本コマンド

これまで学んだ環境設定の基礎知識と基本テクニックを使えば、DOS/Vマシンの環境を構築していくことができますが、実際の環境設定を行うには、さらにCONFIG.SYSやAUTOEXC.BATで使用されるコマンドの意味と使い方を理解しなければなりません。

第*1*章
*CONFIG.SYS*の基本コマンド

DOS/Vの環境構築の最も基本となるのはCONFIG.SYSです。CONFIG.SYSにはDOS/V自身の環境を整える基本的なコマンドと、周辺機器やメモリの環境を構築するデバイスドライバを記述しますが、まずそのCONFIG.SYSの中でも基本となるコマンドから解説していきましょう。

CONFIG.SYSの基本コマンドの設定

まずはCONFIG.SYSの基本コマンドから解説します。

●BUFFERS

形式：BUFFERS＝n,(m)

n＝バッファ数

m＝先読みバッファ数

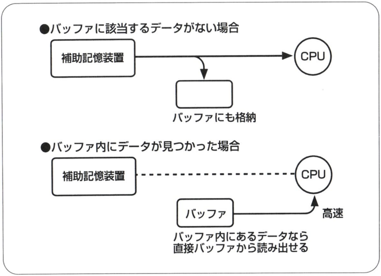
目的：データキャッシングのためのメモリ量を指定する

ハードディスクやフロッピーディスクなどの補助記憶装置は、物理的にデータを読むためのヘッドがデータの記録されている位置に移動しデータを読み出す（書き込む）のですから、どうしてもその処理に時間がかかってしまいます。まったく別のデータを読み出す場合はこれでもいたしかたありませんが、同じデータを２回続けて読み込む場合でもこんな効率の悪いことをやるのもなんとなくしゃくにさわります。

そこで、１回読み出したデータをあらかじめ確保しておいたメモリに蓄えておき、再度同じデータを読み込む時はそのメモリから読み込むような仕組みが考えられました（図11)。このような仕組みを**「バッファ」**あるいは

「キャッシュ」と呼びますが、メモリから直接データを読み出す場合、ハードディスクやフロッピーディスクから読み出す場合に比べ段違いに高速です。

◎図１１　データのキャッシング



「BUFFERS」コマンドは、このデータのバッファに使用するメモリ量を指定するコマンドです。

バッファ数：n

現在のパソコン（コンベンショナルメモリが 640K バイト搭載されたパソコン）指定を省略した場合のデフォルト値は15です。

通常は20～40が適当とされていますが、Windows3.1 の SMARTDRV など別の[1]**キャッシュプログラム**を利用する場合は10程度にしておいた方が効率よくメモリを利用することができます。

バッファにはメモリが割り当てられますので、バッファ一つについて532 バイトのメモリが消費されます。ただし、DOS6 ではバッファ用のメモリがまず HMA 領域に確保されますので、とりあえずコンベンショナルメモリには影響はありません。もしも心配な方は MEM コマンドで確認してみ

[1]「ディスクキャッシュプログラムを組み込む」参照。

てください。

まず、「MEM ↵」として現在のメモリ状況を確認します（画面27）。次に現在のバッファ数＋１のバッファ数を BUFFERS で指定し、パソコンを再起動させてから再度「MEM ↵」でメモリ状況を確認します。

◎画面 27　現在のメモリ状況を確認する

```
C:¥>MEM

メモリーのタイプ      合計  =  使用済 +  使用可
----------------  -------  -------  -------
基本                 640K     101K     539K
UMB                   63K      53K      10K
アダプタ RAM/ROM       0K       0K       0K
XMS* メモリー      10,164K   2,805K   7,359K
----------------  -------  -------  -------
システム全メモリ   10,867K   2,959K   7,908K

1MB 以下全メモリ     703K     154K     549K

全 EMS メモリー                  10,544K (10,797,056 バイト)
使用可能 EMS* メモリー            7,552K  (7,733,248 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ       539K (551,616 バイト)
使用可能最大 UMB メモリー            5K   (5,968 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

ここの値が小さくなる

ここで「コンベンショナル」の空き容量が減ったのであれば、バッファがコンベンショナルメモリに設定されていますので、BUFFERS の値を減らさなければなりません。

先読みバッファ数：m

補助記憶装置からデータを読み出す場合、ある決まった単位（セクタ単位）のデータをキャッシュするメモリ（先読みバッファ）を確保します。

必要なデータがディスク上に連続して記録されていて、かつそのデータに対して連続的な処理が必要な場合は有効ですが、かな漢字変換機能の辞書やデータベースソフトなど、さまざまな領域のデータをアクセスする処理にはあまり効果がありません。

指定を省略するとデフォルト値で「０」となりますので、通常は指定を省略してもかまいません。なお、SMARTDRV などのキャッシュプログラ

ムは**先読みバッファ機能**も備えていますので、これらのプログラムを利用する時は先読みバッファは指定しないようにしましょう。

● FILES

形式：FILES＝n

目的：同時にオープンできるファイル数を指定する

同時にオープンするファイルの数を指定します。MS-DOS だけを利用するなら20もあれば十分ですが、Windows3.1 を利用するとなると最低でも30を指定することになっています。

なお、MS-DOS では標準デバイスドライバも「ファイル」として扱われますので、最低でも５つは必要です。

● DOS

形式：DOS＝HIGH or DOS＝UMB or DOS＝HIGH,UMB

目的：DOS の一部を HMA にロードする
UMB を DOS で管理できるようにする

HIGH

MS-DOS のシステムの一部を HMA メモリに読み込みます。このオプションを指定しておかないと、MEM コマンドを実行したときに「DOS は HMA 領域に常駐しています」というメッセージが表示されません。

UMB

UMB メモリを DOS が管理できるようにします。この指定を行わないと、UMB メモリを MS-DOS から利用することができません。

● COUNTRY

形式：COUNTRY＝国別コード、コードページ、国別情報ファイル

目的：国別情報を読み込む

PC-DOS では、COUNTRY コマンドによって国別情報を読み込みます。たとえば、日本の国別情報を読み込むには次のように指定します。

```
COUNTRY=081,932,C:¥DOS¥COUNTRY.SYS↵
```

なぜわざわざ国別情報を読み込まなければならないかというと、これはもうPC/AT互換機の国際性のせいと言わざるを得ません。PC/AT互換機が世界中で使われているので、その国々にあった通貨の表記や日付の表記形式に合わせる必要があるからです。

たとえば、日本では通貨は「¥」ですし、日付は「1995-08-01」というようになりますが、アメリカでは「$」と「08-01-1995」という形式になります。

また、DOS/Vでは**CHEVコマンド**で日本語環境と英語環境を切り替えることができますが、COUNTRYコマンドで国別情報を読み込んでおかなければこの**CHEV**コマンドも使えません。主要な国の国別コードを表にしておきましたので参考にしてください（表1）。

◎表１　主要国別コード一覧

国名	国コード	コードページ	キーボードコード
日本	081	932,437	jp
アメリカ合衆国	001	437,850	us
フランス	033	850,437	fr
ドイツ	049	850,437	gr
イギリス	044	850,437	uk
イタリア	039	437,850	it

なお、MicrosoftのMS-DOSではCOUNTRYコマンドではなく、AUTOEXEC.BATで**NULFUNC.EXE**を利用しますので、AUTOEXEC.BATのところで解説します。

● SHELL

形式：SHELL＝COMMAND.COM/Pシェルの再読み込み先　/E:環境領域のサイズ

目的：シェルを指定する

ユーザーが入力したコマンドなどを解析し、OS がわかるような言葉で伝えるのがシェルの役目です。MS-DOS では **SHELL** として通常「COMMAND.COM」を指定します。だいたい次のようになっているでしょう。

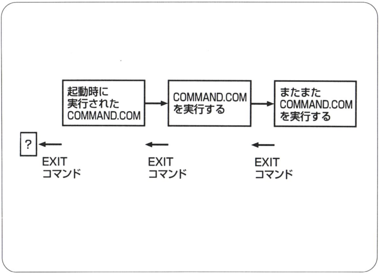
```
SHELL = C:¥DOS¥COMMAND.COM □/P □ C:¥DOS↵
```

/P

このスイッチは SHELL コマンドのオプションスイッチではなく、**COMMAND.COM** のスイッチです。この/P により COMMAND.COM を終了してしまうことを防ぎます。というのも、COMMAND.COM は「EXIT」というコマンドで終了させることができてしまうからです。

通常は「EXIT」コマンドなんか使いませんから、ほとんど問題はないのですが、COMMAND.COM からさらに COMMAND.COM を呼出した場合は EXIT コマンドを利用することとなりますので、念のためこのオプションを付けておいた方がいいでしょう（図12)。

◎図 12　EXIT コマンドで COMMAND.COM を終了させていく



ところで、SHELL である COMMAND.COM を終了させてしまうとどうなるかというとやってみればわかります。この「/P」をつけないでパソ

コンを起動し、DOS のプロンプトから「EXIT ↵」と実行してみてください。みごとにハングアップします。

シェルの再読み込み先

COMMAND.COM は最小限の部分をメモリに常駐させますが、その最小常駐部分も破壊されてしまうことがあります。その場合、パソコンをちゃんと利用するためには再度 COMMAND.COM をロードしなければなりません。どこのディレクトリから COMMAND.COM を再ロードするかを指定します。

環境領域のサイズ E:

AUTOEXEC.BAT で設定する**「環境変数」**（後ほど解説します）は、**環境領域**と呼ばれるメモリに記憶されます。この環境領域はデフォルト（初期値）で256バイト用意されていますが、設定が複雑になると256バイトを超えてしまうこともありますし、よりメモリを節約したい場合は実際に必要な分だけ環境領域として確保すればいいのですから、その場合はこのオプションによって環境領域の大きさを指定します。ただし、最低が160バイト、最高が32768バイトです。

環境変数のサイズが、用意されている環境領域を超えると起動時にエラーメッセージが表示されますのでわかりますが、ピッタリのサイズにするには若干テクニックが必要です。まず次のようにして環境変数をファイルに書き出してください。

```
SET > SET_SIZE.DAT↵
```

さらに、「DIR □ SET_SIZE.DAT ↵」としてそのファイルのサイズを見ます（画面28）。このファイルサイズが環境変数のサイズとなります（実際は少々違います）。念のためにこの SET_SIZE.DAT の内容を見てみると画面29のようになります。

◎画面 28　環境変数のサイズを調べる

```
C:¥>SET > SET_SIZE.DAT

C:¥>DIR SET_SIZE.DAT

 ドライブ C のボリュームは TP-C          です
 ボリューム・シリアル番号は 1ED6-0517 です
 ディレクトリーは  C:¥

SET_SIZE DAT          170 95-07-22   11:39
        1 個              170 バイトのファイルがあります
                   26,923,008 バイトが使用可能です

C:¥>_

    英数 半角   R
```

◎画面 29　SET_SIZE.DAT の内容

```
C:¥>TYPE SET_SIZE.DAT
COMSPEC=C:¥DOS¥COMMAND.COM
PROMPT=$P$G
PATH=C:¥WINDOWS;C:¥DOS;C:¥THINKPAD¥;C:¥IBMVESA;C:¥EZGOING;C:¥LLRA;C:¥DOS;
TEMP=C:¥DOS
IBMAV=C:¥DOS
SPEECHDATA=C:¥SPEECHW¥BIN

C:¥>_

    英数 半角   R
```

ところで、サイズを調べる場合、**「COMSPEC」**という見慣れない一行が表示されるかもしれません。これは「SHELL」コマンドを使用するとそ

こで指定されたシェルが自動的に「COMSPEC」という環境変数に登録されるからなのです。したがって、AUTOEXEC.BATで「**SET␣COMSPEC**」という設定を行ってなくても、ここでやったように「SET」コマンドで表示すると「COMSPEC＝C:¥DOS¥COMMAND.COM」というように表示されるのです。

なお、ここでピッタリ一致させるのもいいでしょうが、若干余裕を持たせておいた方が安全です。というのも、アプリケーションで環境変数を自動的にセットしたりするものもあるからです。本当に安全を期すなら、ここで調べたサイズに128バイトプラスしておけばいいでしょうが、そこまで余裕を見なくても、50バイトもあれば問題はないでしょう。

●INSTALL

形式：INSTALL＝常駐プログラム

目的：常駐プログラムをCONFIG.SYSで組み込みます

通常、常駐型のプログラムはAUTOEXEC.BATで組み込みますので、このINSTALLコマンドは利用する機会がほとんどありません。

INSTALLコマンドではいったん組み込んだプログラムの常駐を解除できなくなりますし、そもそもAUTOEXEC.BATで組み込んだ方が自由度が高いですし、また安全でもあります。

ただし、唯一の例外として、**IBMMKKV.EXE**という連文節変換のかな漢字変換機能はこのINSTALLコマンドで組み込まなければなりません。

ということで、日本語変換プログラムにIBMの**IBMMKKV**を利用しないのであれば、INSTALLコマンドはほとんど必要ないということです。

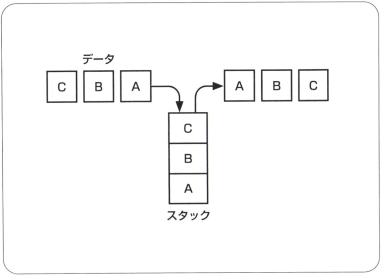
●STACKS

形式：STACKS＝n,m

目的：スタックの数を指定します

パソコンではデータを**スタック**と呼ばれるメモリに退避させたり、そこから呼び出したりといった作業が行われることがあります（図13）。STACKSコマンドではその際に使用されるスタック数とスタックのサイズを指定することができます。

◎図１３　スタックの役割



スタック数：n

使用するスタック数を指定します。指定できる範囲は８〜64ですが、まったく使用しない場合は０を指定することができます。なお、デフォルトでは９です。

スタックサイズ：m

使用するスタックのサイズをバイト単位で指定します。32〜512バイトの範囲内が有効ですが、やはり０を指定することも可能です。なお、デフォルトは128となっています。

通常DOSのみで使用する場合は、**「STACKS＝0,0」**というように指定しスタックを使わない設定にした方がメモリの節約になります。ただし、この設定で**「スタック不足」**とか**「StackOverflow」**といったスタック不足のメッセージが表示された場合、またはWindowsの動作が不安定になってしまった場合は、たとえば**「STACKS＝9,256」**というように、そのトラブルが解消されるだけのスタックとスタックサイズを指定しなければなり

ません。

●LASTDRIVE

形式：LASTDRIVE =最終ドライブ名

目的：利用できる最終ドライブ名を指定する

MS-DOS では自動的に A～E までの５つのドライブ名が用意されていますので、利用するドライブが５ドライブ以内なら LASTDRIVE コマンドを使用する必要はありません。また、ハードウェア的に接続されているドライブは、５つを超えてしまっても自動的に割り当てられますから、この場合も LASTDRIVE コマンドで使用できるドライブ数を指定する必要はありません。

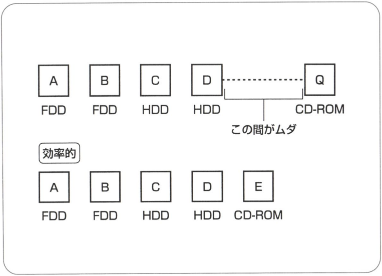
この LASTDRIVE が一番必要な場面で最も身近な例というと、CD-ROM にドライブを割り当てる場合でしょう。また、**SUBST コマンド**で作成したドライブ、または**ネットワークドライブ**などを使用する場合も、使用するドライブ数に応じた最終ドライブ名を指定する必要があります。

最終ドライブ名には利用する最後のドライブ名を指定します。したがって、Q ドライブまで利用するのであれば「LASTDRIVE = Q」となるわけです。これで、A から Q までの16ドライブを利用できるようになりますが、必要以上のドライブ数を指定するとそれだけメモリを消費することとなりますから、必要最低限のドライブ数にとどめておいた方がいいでしょう。

たとえば、CD-ROM には Q ドライブが割り当てられることが多いのですが、なにもわざわざ Q ドライブにする必然性はなく、適当にわかりやすいドライブ名を割り当てておけばいいのです。たとえば、ハードディスクが２ドライブという構成なら、CD-ROM は E ドライブに割り当てることができますので、それなら LASTDRIVE コマンドは必要ないということになります。これならメモリも無駄になりません（図14）。

なお、CD-ROM のドライブの割り当てはその CD-ROM 特有の手順がありますが、通常は AUTOEXEC.BAT に記述する **MSCDEX コマンド**で行います。

◎図14　CD-ROMドライブのドライブ割り当てを考える



● REM

形式：REM

目的：CONFIG.SYSの内容を行単位で無効とする

CONFIG.SYSの内容で、一時的に不要な行や実行したくない行の先頭にREMコマンドを付ければ、その行はコメント行とみなされ実行されません。たとえば、「**STACKS = 9,256**」を無効にするなら次のようになります。

```
REM □ STACKS = 9,256 ↵
```

この性質を利用して、CONFIG.SYS内にコメントを書くこともできます（画面30）。

◎画面 30　CONFIG.SYS 内にコメントを書く

```
DEVICE=C:¥DOS¥EMM386.EXE RAM
DEVICEHIGH=C:¥DOS¥SETVER.EXE
DEVICEHIGH=C:¥DOS¥$DISP.SYS /HS=OFF
DEVICEHIGH=C:¥DOS¥$IAS.SYS
rem **************
rem PCMCIA DRIVER
rem **************
rem DEVICE=C:¥EZGOING¥IBMDSS01.SYS /M
rem DEVICE=C:¥EZGOING¥IBMDOSCS.SYS
rem DEVICE=C:¥EZGOING¥DICRMU01.SYS
rem DEVICE=C:¥EZGOING¥$ICPMDOS.SYS
DEVICE=C:¥DOS¥POWER.EXE
rem DEVICE=C:¥EZGOING¥AUTODRV.SYS C:¥EZGOING¥AUTODRV.INI
rem DEVICE=C:¥EZGOING¥DISKDRV.SYS
DEVICEHIGH=C:¥DOS¥$PRN.SYS
DEVICEHIGH=C:¥DOS¥ANSI.SYS /X
DEVICEHIGH=C:¥DOS¥$FDD12.SYS
REM DEVICEHIGH=C:¥DOS¥RAMDRIVE.SYS
INSTALL=C:¥DOS¥IBMMKKV.EXE /M=S /Z=4 /C /L /J=90 /S=C:¥DOS¥MULTDICT.PRO /U=C:¥$U
SRDICT.DCT
STACKS=9,256
LASTDRIVE=Z

C:¥>_
    英数 半角   R
```

第２章
*AUTOEXEC.BAT*の基本コマンド

CONFIG.SYSでDOS/Vの基本的な環境を構築したら、あとはAUTOEXEC.BATでアプリケーショプログラムが快適に動作するように環境構築を行わなければなりません。

このAUTOEXEC.BATでもCONFIG.SYSと同じように、DOS/V自身のために実行するコマンドと周辺機器やアプリケーションプログラムのために実行するコマンドに分かれますが、AUOEXEC.BATではCONFIG.SYSのときのように明確な区別はつきませんので、一挙に解説していきましょう。

AUTOEXEC.BATの基本コマンド

AUTOEXEC.BATで使用するコマンドは、AUTOEXEC.BATでのみ使用されるものではなく、他のバッチファイルで使ったりコマンドラインから直接入力したりすることも可能です。

●SET

形式：SET　環境変数

目的：環境変数をセットする

環境変数は単に設定しただけでは何も意味がありませんが、その環境変数をセットしておくことでプログラムの動作環境を整えるのです。

このSETコマンドはユーザーが直接利用する機会はあまりないのですが、アプリケーションソフトをインストールしたりすると、そのインストーラが自動的にSETコマンドを追加することもあります。たとえば、

Thinkpad230Cs の AUTOEXC.BAT には次のように SET コマンドが使われています。

```
SET □ COMSPEC = C:¥DOS¥COMMAND.COM
SET □ TEMP = C:¥DOS
SET □ IBMAV = C:¥DOS
SET □ SPEECHDATA = C:¥SPEECHW¥BIN
```

なお、SET コマンドに引数をつけないと現在の環境変数の一覧が表示されます（画面31）。

◎画面３１　SET コマンドで環境変数を表示させることができる

```
C:¥>SET
COMSPEC=C:¥DOS¥COMMAND.COM
PROMPT=$P$G
PATH=C:¥WINDOWS;C:¥DOS;C:¥THINKPAD¥;C:¥IBMVESA;C:¥EZGOING;C:¥LLRA;C:¥DOS;
TEMP=C:¥DOS
IBMAV=C:¥DOS
SPEECHDATA=C:¥SPEECHW¥BIN

C:¥>_




    英数 半角  R
```

● ECHO

形式：ECHO

：ECHO ON | OFF

：@ECHO OFF

目的：画面にメッセージを表示したり、制御したりする

ECHO コマンドはさまざまな使い道がありますが、使い方は比較的やさしく使いやすいコマンドでしょう。バッチファイルの中で使われることがほ

とんどです。次のような簡単なバッチファイルを作って実験してみるとその機能がよくわかります。

```
DIR □/W ↵
```

とりあえずこの1行だけのバッチファイルを作成してください。名前はなんでもいいですが、ここでは「E-1.BAT」としておきましょう。エディタやワープロを起動するまでもなく、次のような操作で簡単に作成できます。

```
COPY □ CON □ E-1.BAT ↵
DIR □/W ↵
```

ここまで入力したら CTRL + Z を押すとファイルが作成できます。その E-1.BAT を実行してみましょう（画面32)。

◎画面32　E-1.BATの実行結果

```
C:¥>E-1

C:¥>DIR /W

 ドライブ C のボリュームは TP-C        です
 ボリューム・シリアル番号は 1ED6-0517 です
 ディレクトリーは  C:¥

[AIR]          [AS]           [ATOK]         [ATOK9]        [BAT]
[CAPIT]        [CW]           [DOS]          [ED]           [EXCEL5]
[EZ]           [FM]           [FONT]         [GAME]         [HT]
[IBMVESA]      [ICON]         [ICONDOIT]     [ICONMAKE]     [JSLIB]
[JSTOOL]       [LHA]          [MDEV]         [MIW]          [MSDOS]
[MSOFFICE]     [MYFAX]        [NU]           [NW]           [ORGANIZE]
[PKARC]        [POWER2]       [PTF310B]      [QEMM]         [OV10]
[SFW]          [SKW]          [SYS]          [SYSBAK]       [TAROWIN6]
[TBACK]        [TEST]         [TP]           [VCAP]         [WIN32APP]
[WINDOWS]      AUTOEXEC.BAT   COMMAND.COM    CONFIG.SYS     E-1.BAT
KAN.BAT        SET_SIZE.DAT
        52 個            60,490 バイトのファイルがあります
                     26,939,392 バイトが使用可能です

C:¥>
C:¥>_
   英数 半角   R
```

ECHO OFF

コマンドのエコーバックを表示しません。このコマンドの実行結果を見るためには、さきほどのバッチファイル「E-1.BAT」にこの「ECHO OFF」を加えたバッチファイルを作成します。なお、通常「ECHO

OFF」はバッチファイルの先頭に記述します。

新しいバッチファイルの内容は次のようにしてください。

```
ECHO □ OFF↵
DIR □/W↵
```

名前は「E-2.BAT」にしておきましょう。作成方法は「E-1.BAT」と同じですが、ただ「COPY CON E-2.BAT」としなければなりません。

作成したら「E-2.BAT」を実行してみましょう（画面33）。

◎画面33　ECHO OFFでは

```
C:¥>E-2

C:¥>ECHO OFF

 ドライブ C のボリュームは TP-C        です
 ボリューム・シリアル番号は 1ED6-0517 です
 ディレクトリーは  C:¥

[AIR]          [AS]           [ATOK]         [ATOK9]        [BAT]
[CAPIT]        [CW]           [DOS]          [ED]           [EXCEL5]
[EZ]           [FM]           [FONT]         [GAME]         [HT]
[IBMVESA]      [ICON]         [ICONDOIT]     [ICONMAKE]     [JSLIB]
[JSTOOL]       [LHA]          [MDEV]         [MIW]          [MSDOS]
[MSOFFICE]     [MYFAX]        [NU]           [NW]           [ORGANIZE]
[PKARC]        [POWER2]       [PTF310B]      [OEMM]         [QV10]
[SFW]          [SKW]          [SYS]          [SYSBAK]       [TAROWIN6]
[TBACK]        [TEST]         [TP]           [VCAP]         [WIN32APP]
[WINDOWS]      AUTOEXEC.BAT   COMMAND.COM    E-2.BAT        CONFIG.SYS
E-1.BAT        KAN.BAT        SET_SIZE.DAT
        53 個           60,508 バイトのファイルがあります
                    26,935,296 バイトが使用可能です
C:¥>_

   英数 半角   R
```

このように、コマンドを実行する「DIR □/W」そのものが表示されなくなります。また、プロンプト状態でキーボードから「ECHO □ OFF↵」と入力すると、コマンドプロンプトが消えます。

■ @ECHO OFF

ところがあいかわらず「ECHO OFF」自身は表示されてしまいます。このままだとなかなか間抜けです。その「ECHO OFF」自身も消してしまうためには、頭に「@」をつけます。つまり「ECHO　OFF」を「@ECHO OFF」とするわけです。またまた次のようなバッチファイル（今度はE-3.

BAT）を作りましょう。

```
@ECHO □ OFF↵
DIR □/W↵
```

さて、実行結果は画面34のようになります。

◎画面34　ECHO OFFも消し去る

```
C:¥>E-3

 ドライブ C のボリュームは TP-C         です
 ボリューム・シリアル番号は 1ED6-0517 です
 ディレクトリーは  C:¥

[AIR]           [AS]            [ATOK]          [ATOK9]         [BAT]
[CAPIT]         [CW]            [DOS]           [ED]            [EXCEL5]
[EZ]            [FM]            [FONT]          [GAME]          [HT]
[IBMVESA]       [ICON]          [ICONDOIT]      [ICONMAKE]      [JSLIB]
[JSTOOL]        [LHA]           [MDEV]          [MIW]           [MSDOS]
[MSOFFICE]      [MYFAX]         [NU]            [NW]            [ORGANIZE]
[PKARC]         [POWER2]        [PTF310B]       [QEMM]          [QV10]
[SFW]           [SKW]           [SYS]           [SYSBAK]        [TAROWIN6]
[TBACK]         [TEST]          [TP]            [VCAP]          [WIN32APP]
[WINDOWS]       AUTOEXEC.BAT    E-3.BAT         COMMAND.COM     E-2.BAT
CONFIG.SYS      E-1.BAT         KAN.BAT         SET_SIZE.DAT
     54 個              60,527 バイトのファイルがあります
                    26,931,200 バイトが使用可能です
C:¥>_


   英数 半角   R
```

通常AUTOEXEC.BATの先頭に「@ECHO OFF」が入っていますので、AUTOEXEC.BATが実行されるときにそのコマンドがエコーバックされないのです。

ECHO ON

「@ECHO OFF」コマンドでエコーバックを消すようにしていても、一部だけエコーバックが欲しいこともあります。その場合はそのコマンドの前に「ECHO ON」を挿入しておきましょう。次のような「E-4.BAT」を作りましょう。結果は画面35のようになります。

```
@ECHO □ OFF↵
ECHO □ ON↵
```

```
DIR □/W⏎
```

◎画面 35　ECHO ON の挿入

```
C:¥>E-4

C:¥>DIR /W

 ドライブ C のボリュームは TP-C        です
 ボリューム・シリアル番号は 1ED6-0517 です
 ディレクトリーは  C:¥

[AIR]          [AS]           [ATOK]         [ATOK9]        [BAT]
[CAPIT]        [CW]           [DOS]          [ED]           [EXCEL5]
[EZ]           [FM]           [FONT]         [GAME]         [HT]
[IBMVESA]      [ICON]         [ICONDOIT]     [ICONMAKE]     [JSLIB]
[JSTOOL]       [LHA]          [MDEV]         [MIW]          [MSDOS]
[MSOFFICE]     [MYFAX]        [NU]           [NW]           [ORGANIZE]
[PKARC]        [POWER2]       [PTF310B]      [QEMM]         [QV10]
[SFW]          [SKW]          [SYS]          [SYSBAK]       [TAROWIN6]
[TBACK]        [TEST]         [TP]           [VCAP]         [WIN32APP]
[WINDOWS]      AUTOEXEC.BAT   E-3.BAT        COMMAND.COM    E-2.BAT
CONFIG.SYS     E-1.BAT        KAN.BAT        E-4.BAT        SET_SIZE.DAT
       55 個            60,554 バイトのファイルがあります
                    26,927,104 バイトが使用可能です

C:¥>_
    英数 半角   R
```

「ECHO ON」のおかげで無事「DIR □/W」が表示されました。なお、コマンドラインで「ECHO □ ON⏎」を実行すると「ECHO OFF」コマンドで消した**プロンプト**を再度表示させることができます。

ECHO

現在の ECHO の状態を知りたい場合は、ON も OFF もつけずに ECHO コマンドを実行してください（画面36）。

◎画面 36　ECHO コマンドで ECHO の状態を知る

```
C:¥>ECHO
ECHO は <on> です

C:¥>ECHO OFF
ECHO
ECHO は <off> です

ECHO ON

C:¥>ECHO
ECHO は <on> です

C:¥>_




英数 半角　R
```

ECHO の状態を知ったとしても特に便利ということもありません。

■ ECHO メッセージ

ECHO コマンドではメッセージを画面に表示させることもできるのです。今度は「E-5.BAT」でしたね。内容は次のようにしてください。

```
@ECHO □ OFF ↵
ECHO こんなメッセージも表示することができます。
```

実行結果は画面37のようになりますが、このテクニックは独自のメニューなどを作成する時に役立ちます。

◎画面 37　ECHO でメッセージを表示させる

```
C:¥>TYPE E-5.BAT
@ECHO OFF
ECHO こんなメッセージも表示することができます。

C:¥>E-5
こんなメッセージも表示することができます。
C:¥>_




英数 半角   R
```

■ その他

ECHO コマンドは、DOS の[2]**リダイレクト**と呼ばれる機能を利用してテキストファイルに文字列を追加したりすることもできますが、一般的な利用方法ではありませんのでここでは解説しません。とにかくここまでの利用方法をマスターすれば、環境構築には困らないはずです。

● PATH

形式：PATH =パスを通すディレクトリ

目的：コマンドサーチパスを設定する

環境変数の中でも最も頻繁に利用され、かつ重要な働きをするのが「PATH」コマンドです。

DOS/V でのコマンドは原則としてカレントディレクトリに対してのみ有効です。したがって、何かプログラムを実行しようとした場合、カレントデ

[2]「<」や「>」を使って出力先を変更すること。通常は画面に出力されるものを、ファイルやプリンタに出力することもできる。

ィレクトリに実行したいプログラムファイルがなければ、実行することができません。これではいくらなんでも不便なので、「PATH」コマンドでプログラムをサーチするディレクトリを指定することができるようになっています。つまり、PATH コマンドで指定したディレクトリにあるプログラムなら、カレントディレクトリがどこであろうが実行できるのです。

PATH コマンドでは複数のディレクトリを指定することも可能で、その場合は各ディレクトリをセミコロン「；」で区切って指定します。ただし、全体の長さが半角で127文字以内でなければなりません。それを越えた部分は無効となります。

たとえば、「C:¥」「C:¥DOS」「C:¥WINDOWS」という３つのディレクトリを PATH 指定する場合は次のようになります。

```
PATH = C:¥;C:¥DOS;C:¥WINDOWS;↵
```

このように PATH コマンドでディレクトリが設定されている場合、ユーザーが何かコマンドを入力すると、COMMAND.COM は次の順序で対応したプログラムをサーチします。

１．内部コマンド

「DIR」とか「COPY」といった内部コマンドであれば、そのコマンドを実行します。

２．カレントディレクトリ

内部コマンドではない場合は、現在のディレクトリから入力された名前に対応するプログラムを実行します。

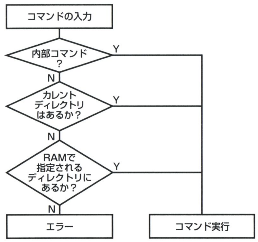
３．PATH コマンドで指定されたディレクトリ

カレントディレクトリに対応するプログラムがない場合は、環境変数「PATH」で指定されたディレクトリ内を順次検索します。

検索する順序は、PATH コマンドで指定されている順序です。

たとえば、現在カレントディレクトリが A:¥で PATH コマンドに「PATH = C:¥;C:¥DOS;C:¥WINDOWS;」というように指定されている場合、まず「A:¥」を検索し続いて「C:¥」「C:¥DOS」を、最後に「C:¥WINDOWS」をという具合です。もちろん、A:¥に目的のプログラムファイルがあれば、そのままプログラムは実行されます（図15）。

◎図15　PATHコマンドのプログラム検索順序



したがって、PATHコマンドでは頻繁に利用するディレクトリをできるだけ先頭にもってきた方が、マシン全体の実効速度がほんのちょっとですが向上します。

これを厳密に利用してしまおうというのなら、プログラムを起動するバッチファイルでそのプログラムがあるディレクトリをPATHの先頭に記述するようにしむければいいわけです。環境変数は前と後ろを%で囲えば参照することができますから、ちょっと工夫すれば簡単です。実験してみましょう。次のようなバッチファイルを作成してください。名前は「P.BAT」とでもしておきましょう

```
@ECHO □ OFF
ECHO 最初の PATH 設定
PATH
SET □ OLDPATH =%PATH%
PATH = C:¥TEST;%PATH%
```

```
ECHO TEST が先頭になっています
PATH
PATH =%OLDPATH%

ECHO 元に戻しました
PATH
```

実行結果は画面38のようになります。

◎画面 38　PATH の操作

```
C:¥>P
最初のPATH設定
PATH=C:¥WINDOWS;C:¥DOS;C:¥THINKPAD¥;C:¥IBMVESA;C:¥EZGOING;C:¥LLRA;C:¥DOS;
TESTが先頭になっています
PATH=C:¥TEST;C:¥WINDOWS;C:¥DOS;C:¥THINKPAD¥;C:¥IBMVESA;C:¥EZGOING;C:¥LLRA;C:¥DOS
;
元に戻しました
PATH=C:¥WINDOWS;C:¥DOS;C:¥THINKPAD¥;C:¥IBMVESA;C:¥EZGOING;C:¥LLRA;C:¥DOS;
C:¥>_




    英数 半角   R
```

PATH 指定の効果を具体的に体感するために、次のような手順を実行して下さい。

1．C:¥にカレントディレクトリを移動する

2．PATH を確認する

「PATH ⏎」として現在どのディレクトリに PATH 設定がされているかを確認してください（画面39）。

◎画面 39　PATH の設定を確認する

```
C:¥>PATH
PATH=C:¥WINDOWS;C:¥DOS;C:¥THINKPAD¥;C:¥IBMVESA;C:¥EZGOING;C:¥LLRA;C:¥DOS;

C:¥>_




    英数 半角  R
```

3．何か DOS のコマンドを実行してみる

PATH 設定を確認すると DOS のディレクトリには PATH が通っていますので、DOS のディレクトリ内にあるプログラムなら、カレントディレクトリが C:¥でも実行できるはずです。無害なところで SETVER でも実行してみて下さい。画面40のように SETVER のバージョン報告が表示されます。

◎画面 40　SETVER のバージョン報告

```
IBMCACHE.COM    3.40
WIN100.BIN      3.40
HITACHI.SYS     4.00
MSCDEX.EXE      4.00
NET.EXE         4.00
NETWKSTA.EXE    4.00
BAN.EXE         4.00
BAN.COM         4.00
DD.EXE          4.01
DD.BIN          4.01
REDIR.EXE       4.00
SYQ55.SYS       4.00
SSTDRIVE.SYS    4.00
ZDRV.SYS        4.01
ZFMT.SYS        4.01
TOPSRDR.EXE     4.00
NETBEUI.DOS     5.00
CIP_MAIN.EXE    5.00
BACKUP.COM      5.00
AD-DOS.COM      5.00
KKCFUNC.SYS     5.00
ADDDRV.EXE      6.10

C:¥>_
    英数 半角   R
```

4．PATH 指定をクリアする

「PATH;↵」と入力して下さい（セミコロン「；」を忘れずに)。これで PATH 指定がすべてクリアされているはずです。「PATH ↵」を実行して「パスが定義されていません」というメッセージが表示されることを確認して下さい。

5．SETVER を実行する

再度「SETVER」を実行してみて下さい。今度は PATH が通っていませんので、「コマンドまたはファイル名が正しくありません」というメッセージが表示されるはずです（画面41)。

◎画面 41　プログラムが見つからないエラーメッセージ

```
C:¥>PATH;

C:¥>PATH
パスが定義されていません


C:¥>SETVER
コマンドまたはファイル名が正しくありません

C:¥>_









    英数 半角   R
```

6. PATH を設定する

DOS のディレクトリに PATH を通しましょう。「PATH = C:¥DOS ⏎」と入力して下さい。これでまたまた SETVER を実行してみて、最初と同じように SETVER バージョン報告が表示されたら PATH の設定は成功です。

PATH 指定を正確に行わないと、動くものも動かなくなってしまいます。それほど PATH コマンドは重要ですが、アプリケーションをインストールすると自動的に PATH 設定も行ってくれることが大半ですので、ユーザーは PATH 指定そのものよりも PATH 指定が127文字を越えていないかどうかのみ気を配っておけばいいでしょう。

なお、PATH 指定がどんどん増えてくると、環境変数エリアが足りなくなることがありますので、起動時に表示される「環境空間が足りません」というようなメッセージを見つけたら、CONFIG.SYS の SHELL コマンドで「/E」オプションをつけて環境変数のためのエリアを広げて下さい。

● PROMPT

形式：PROMPT オプション

目的：DOSのプロンプト表示をカスタマイズする

DOSの画面は非常に味気ないものですが、その味気なさにさわやかな風を送り込んでいるのがプロンプトです。

だいたいデフォルトで「C:¥>」というように表示されているようですが、PROMPTコマンドによりこのプロンプトをカスタマイズすることができます。そのオプションは表2を参照して下さい。

◎表２　プロンプトコマンドのオプション

オプション	表示	
$Q	=	等号
$$	$	ドルマーク
$T	21:30:00	現在の時刻
$D	08-01-1995（火）	今日の日付
$P	C:¥DOS	カレントドライブ・ディレクトリ
$V	PC DOS Version J6.30/V	DOSバージョン
$N	C	カレントドライブ
$G	>	不等号
$L	<	不等号
$B	|	パイプ
$H	バックスペースが入力される	バックスペース
$E	エスケープコードが入力される	エスケープコード
$_	改行される	改行

デフォルトではカレントドライブのカレントディレクトリを表示するというようにAUTOEXEC.BATに書かれています（「PROMPT□PG」という一行）が、このオプションを利用してちょっと遊んでみましょう。なお、PROMPTコマンドはコマンドラインから実行しても有効ですので、ここではコマンドラインから手軽にプロンプトをカスタマイズしてみましょう。

■ 現在の日時時刻を表示する

設定：PROMPT DT$G

コマンドプロンプトに現在の日時（$D）時刻（$T：コマンドプロンプトが表示された時刻）を表示することができます（画面42）。

◎画面 42　日付時刻をプロンプトとして表示する

```
C:¥>PROMPT $D$T$G
1995-07-22 (土)22:33:01.19>_




英数 半角　R
```

画面に色をつける

設定：PROMPT $E[31m$P$G

このようにすると、画面に表示される文字がすべて赤になります。プロンプト以外の文字色も変更されますから気を付けて下さい。とくに「PROMPT □$E[30m」などと指定してしまうと、文字色が黒になりますので画面にはなにも表示されなくなります。

デフォルト状態に戻す

単に「PROMPT ↵」を実行すると、デフォルトのコマンドプロンプト（カレントドライブと＞）が表示されます。ただし、$E で指定した色は元の色に戻りませんから、これを戻すには「PROMPT □$E[m」を実行するか、マシンをリセットしてください。なお、デフォルトの設定は「PROMPT □NG」と同じです。

● LOADHIGH (LH)

形式：LOADHIGH 常駐型プログラム

目的：常駐型プログラムを UMB に常駐させます

このコマンドを使用しないと、AUTOEXEC.BAT で組み込む常駐型プログラムがすべてコンベンショナルメモリに常駐してしまいます。よりたくさんの余裕をコンベンショナルメモリにもたせたい場合は、この LH コマンドを使用して下さい。DEVICEHIGH コマンドの常駐型プログラム版です。たとえば、MOUSE.COM を UMB に常駐させたいのであれば、次のように指定します。

```
LH □ C:¥DOS¥MOUSE.COM ↵
```

ただし、プログラムが必要とするサイズ以上の UMB が残っていないと、LH コマンドは無効になってしまい、その常駐型プログラムはコンベンショナルメモリに常駐します。

どちらにしろコンベンショナルメモリに常駐するのであれば、だめでもともとすべての常駐型プログラムに LH コマンドを使っておけばいいでしょう。プログラムがどれくらいの常駐メモリを必要とするのかは、LH コマンドを利用しないでそのプログラムを常駐させ、MEM コマンドでメモリの減り方を調べてみるとわかります。

● REM

形式：REM

目的：行単位で無効とする

AUTOEXEC.BAT 内の不要な行でも、削除するのはちょっと躊躇してしまうというようなこともあります。また、一時的に無効にしたい行もあるでしょう。

このような場合は、その行の先頭に「REM」をつけると CONFIG.SYS のときと同じようにその行を無効にすることができます。また、コメント行としてバッチファイル内にコメントを書くためにも使われます。

DOS5 までは、AUTOEXEC.BAT では「REM」コマンドが無効でしたので、コロン「：」を使っていました。このコロンは本来バッチファイル内のラベルの指定に使われるものでしたので、へたをするとラベルなのかコ

メント行なのかわからなくなってしまうこともありました。が、DOS6 からは晴れて REM コマンドが使えるようになったので、よりわかりやすい AUTOEXEC.BAT が作成できるようになりました。

第 3 章
メモリ・周辺機器を管理するコマンド

CONFIG.SYS の基本コマンドと AUTOEXEC.BAT で環境を整えたとしても、それではまだ機器も使える状態ではありませんし、DOS/V 上でかな漢字変換機能も利用することができません。さらにメモリや周辺機器を管理するドライバで、マシン本体と周辺機器をシステムとして構築しなければなりません。

メモリを管理するデバイスドライバ

一番重要なのがこのメモリを管理するドライバです。何はともあれこのドライバを組み込まないと、以後の作業がスムーズにいかなくなります。

● HIMEM.SYS

形式：HIMEM.SYS　オプション

目的：XMS メモリを使えるようにする

このデバイスドライバによって、1M 以上の XMS メモリ（HMA メモリも含む）を DOS で利用できるようになります。必ず指定しなければ DOS/V は、ほとんどまともに動きません。もちろん、Windows3.1 環境でも必要です。

XMS メモリを管理するところから**「XMS メモリマネージャ」**などと呼ばれることもあります。

通常この HIMEM.SYS は DOS のディレクトリに入っていますので、実際に指定する際は次のようになります。

```
DEVICE = C:¥DOS¥HIMEM.SYS↵
```

組み込む位置ですが、HIMEM.SYSは原則的に他のデバイスドライバに先だって一番最初に組み込まれなければなりません。もちろん、他のデバイスドライバのうち、XMSメモリを利用しないものはこのかぎりではりませんが、その判断も難しいので「HIMEM.SYSが先頭」と憶えておいた方が簡単です。

ただし、DOS5以前ではHIMEM.SYSの前に$FONT.SYSを組み込むか、HIMEM.SYSに「INT15 =」というオプションスイッチ$FONT.SYSを組み込むためのメモリを明けておくといった操作が必要となりますので注意が必要です。

なお、このオプションを含めHYMEM.SYSには若干のオプションがありますが、現在ではほとんど使用されません。

● EMM386.EXE

形式：EMM386.EXE オプション

目的：1．仮想EMSメモリを有効にする

2．UMBを利用できるようにする

EMM386.EXEはHIMEM.SYSで利用できるようになったXMSメモリをEMSメモリとして利用するデバイスドライバです。したがって、必ずHIMEM.SYSの後に記述しなければなりません。

EMM386.EXEには非常に多くのオプションがありますが、一般的に必要なオプションは次の2つだけです。

RAM

このオプションを指定すると、UMBメモリをDOSで利用できるようになります。必ず指定したいオプションです。たとえば次のようになります。

```
DVICE = C:¥DOS¥EMM386.EXE □ RAM↵
```

X

RAMオプションで利用できるようにしたUMBの中で、DOSで利用し

ないメモリ範囲を指定するオプションです。

PCMCIA カードなどを利用するときにはこのオプションが必要になります。その詳細については後ほど解説します。

なお、DOS5 までは EMM386.EXE で使用する EMS メモリのサイズを指定しなければなりませんでした。これは DOS5 の EMM386.EXE が XMS と EMS メモリを完全に別のものとして扱っていたからでした。

DOS6 では XMS と EMS を一括して扱うことができますので、EMS メモリの量を指定する必要がなくなりました。EMS メモリのサイズを指定しなくても、自動的に必要な EMS メモリを割り当ててくれます。MEM コマンドでメモリ状況を見た場合、その点についてのメッセージが表示されています（画面43）。

◎画面 43　EMS メモリについてのメッセージ

```
C:¥>MEM

メモリーのタイプ    合計  =  使用済 +  使用可
----------------  -------   -------   -------
基本                640K      101K      539K
UMB                  63K       53K       10K
アダプタ RAM/ROM      0K        0K        0K
XMS* メモリー     10,164K    2,805K    7,359K
----------------  -------   -------   -------
システム全メモリ  10,867K    2,959K    7,908K

1MB 以下全メモリ    703K      154K      549K

全 EMS メモリー                 10,544K (10,797,056 バイト)
使用可能 EMS* メモリー           7,552K  (7,733,248 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ       539K (551,616 バイト)
使用可能最大 UMB メモリー            6K   (5,968 バイト)
DOS は HMA 領域に常駐しています

C:¥>_                          これがEMSに関するメッセージ
    英数 半角   R
```

● RAMDRIVE.SYS

形式：RAMDRIVE.SYS ディスク容量 セクタサイズ オプション

目的：RAM ドライブを作成する

RAMDRIVE.SYS を組み込むと、メモリを一時的にハードディスクや

フロッピーディスクと同様の補助記憶装置として利用することができます。

ハードディスクがまだ高価で一般的なユーザーにとって遠い存在であった頃は、増設メモリを **RAM ドライブ**としてかな漢字変換機能の辞書などを RAM ドライブに置いて利用していましたが、現在ではハードディスクが普及していますので RAM ドライブはほとんど利用されません。とくに Windows3.1 環境においては、RAM ドライブに割り当てるメモリがあるなら、Windows3.1 用に残しておいた方がいいということになっています。

もちろん、DOS ベースで利用するのでしたら、搭載しているメモリを有効に利用するために RAM ドライブを作成し、ワークエリアに使用したりかな漢字変換機能の辞書を載せたりするといった使用方法もいいでしょう。

ディスク容量

RAM ドライブに割り当てるメモリ量を指定します。ディスク容量は K バイト単位で指定しますが、指定できる範囲は DOS の種類によって違います。

PC-DOS → 4～31,744
MS-DOS → 4～32,767

セクタサイズ

RAM ディスクのセクタサイズを「128」「256」「512」バイトの中から指定することができますが、「セクタ」が何かわからない方は何も指定する必要はありません。デフォルトで512が指定されます。

より高速なアクセスを求めるなら「512」ですし、より効率よく RAM ディスクを利用しようというのであれば「128」がいいでしょう。

エントリ数

「エントリ数」とは、RAM ディスクのルートディレクトリに作成できるファイルとディレクトリの総数のことです。デフォルトでは64ですが、大きな RAM ディスクを作成し多量のファイルをコピーするというような利用方法を考えているのなら、このエントリ数を増やしておかなければなりません。指定できるエントリ数の範囲は2～1024です。

■ メモリ割り当て元

「/A」ではEMSメモリからRAMディスクにメモリを割り当てますが、「/E」ならXMSメモリから直接割り当てられます。一般的には「/E」を指定します。なお、このオプションを指定しないとコンベンショナルメモリからRAMディスクにメモリが割り当てられてしまいます。

たとえば、2Mバイト、256バイト／セクタ、エントリ数＝128、メモリはXMSからという設定でRAMディスクを作成するなら次のように指定します。

```
DEVICE = RAMDRIVE.SYS □ 2048 □ 256 □ 128 □/E↵
```

もちろん「DEVICEHIGH」コマンドを利用することもできます。この指定を複数記述すれば、複数のRAMディスクを作成することも可能です。

作成されたRAMディスクは、ハードディスクやフロッピーディスクと同じように利用することができますが（画面44）、レジューム機能がサポートされていないマシンの場合は電源を切ったり、リセットしたりすると内容が消滅してしまいますので気を付けてください。

◎画面 44　RAM ディスクが作成されている

```
PC DOS Power Manager Version 1.0
Copyright (c) IBM Corporation 1986, 1991-1993

DOS/V プリンター・ドライバ・マネージャー(V1.01)が導入されました。
3.5 型ディスケット駆動機構デバイス・ドライバー($FDD12.SYS Ver 2.02)
(C) Copyright IBM Corporation 1991,1994. All rights reserved.
3.5 型ディスケット駆動機構デバイス・ドライバーが導入されました。
IBM RAMDrive version 3.06 virtual disk E:
    ディスク・サイズ          : 2048K
    セクター・サイズ          : 256 バイト      ── RAMドライブが組み込まれている
    クラスター・サイズ        : 4 セクター
    ディレクトリー項目数      : 128
連文節変換ルーチン (IBMMKKV.EXE Version 6.03)が EMS メモリーに導入されました。
IBM PS/55系日本語プリンター・ドライバー(PRINIBM.COM Ver. 2.00)が導入されました。


PC DOS Version J6.3/V

Revision 0


C:¥>_
    英数 半角   R
```

2カ国語を利用するためのドライバ（MS-DOS）

Microsoft製MS-DOSで**英語モード**と**日本語モード**を利用し、その両方を切り替えられるようにするには、特別なドライバを組み込まなければなりません。

このドライバはPC-DOSにはありませんので、Microsoft製のMS-DOSでのみ必要です。

● BILING.SYS

形式：BILING.SYS

目的：MS-DOS（Microsoft）にバイリンガル環境を提供する

BILING.SYSを組み込むことによって、Microsoft製のMS-DOSでは英語モードと日本語モードを切り替えることができるようになります。

最近では**「バイリンガル」**と言うと女性（しかも若い女性）にのみ使用される形容詞のようになっていますが、MS-DOSでも2カ国語を利用できる環境を「バイリンガル環境」と呼んでいます。

BILING.SYSには特にオプションスイッチなどは用意されてはいませんが、必ず日本語を表示するドライバ（JFONT.SYS、JDISP.SYS）などの前に組み込まなければなりません。位置としては、HIMEM.SYSとEMM386.EXEの直後ということになります。

```
DEVICE = C:¥DOS¥HIMEM.SYS ↵
DEVICE = C:¥DOS¥EMM386.EXERAM ↵
DEVICE = C:¥DOS¥BILING.SYS ↵
```

日本語を表示するドライバ

DOS/V では日本語をソフトウェア的に表示していますので、その表示をつかさどる２つのデバイスドライバが必要です。

●$FONT.SYS（PC-DOS）/JFONT.SYS（MS-DOS）

形式：$FONT.SYS　オプション
　　　JFONT.SYS　オプション
目的：フォントデータをフォントファイルから読み込みメモリに展開する。

DOS/V では欠かせないデバイスドライバです。もちろん、日本語を使用しないのであれば必要ありませんが、Windows 環境（DOS/V 用の Windows3.1）ではやはり必須です。

組み込み位置

フォントドライバは、必ず次に解説する**ディスプレイドライバ**よりも前にフォントドラバを組み込まなければなりません。位置としては「EMM386.EXE」とディスプレイドライバ「$DISP.SYS」または「JDISP.SYS」の間ということになります。

DOS5 までのフォントドライバは、int15 という特別なアクセス方法で XMS メモリを使っていたので、必ず HIMEM.SYS の前に組み込むか HIMEM.SYS にフォントドライバ用のオプションを付けていましたが、DOS6 からはその心配がなくなりました。

オプション

フォントドライバのオプションは、フォントが格納されているディレクトリを指定する「/P」です。たとえば、C ドライブの¥FONT というディレクトリにフォントファイルがあるなら、次のように指定してください。

```
DEVICEHIGH =$FONT.SYS □/P = C:¥FONT ↵     (PC-DOS)
DEVICEHIGH = JFONT.SYS □/P = C:¥FONT ↵    (MS-DOS)
```

また「/24＝ON」というように24ドットフォントを使用するためのスイッチをつけることもありますが、通常のシステムでしたら必要のないオプションです。

●$DISP.SYS (PC-DOS) /JDISP.SYS (MS-DOS)

形式：$DISP.SYS　オプション
**　　　JDISP.SYS　オプション**

目的：日本語（全角文字）を表示できるように画面モードを設定する。

このドライバによって RAM に読み込んだ全角フォントを画面に表示できるようにするのです。

組み込み位置

ディスプレイドライバの組み込み位置は、フォントドライバの後ろであればとくに制限はありません。もっとも、だいたいフォントドライバのすぐ後ろに組み込むことが多いようです。

オプション

DOS/V では使用するビデオカードにより、**ハードウェアスクロール**を行うと画面が正常に表示されない場合もあります。その時はハードウェアスクロールを制限したり停止したりするオプションで対処することができます。

オプションは「/HS＝」です。$DISP.SYS を例にとると次のようになります。

```
DEVICEHIGH = C:¥DOS¥$DISP.SYS [ /HS = ON]↵
                              ハードウェアスクロール
DEVICEHIGH = C:¥DOS¥$DISP.SYS  /HS = LC ↵
                              ハードウェアスクロール（低速）
DEVICEHIGH = C:¥DOS¥$DISP.SYS  /HS = OFF ↵
                              ハードウェアスクロール OFF
```

何もオプションを指定しない場合が一番高速で、「/HS＝OFF」が最も低速（相当遅い）です。「/HS＝LC」がその中間ということになります。まず何もオプションをつけずに起動してみて、画面の調子がおかしいようで

したら「/HS＝LC」をつけてみます。それでもだめでしたら「/HS＝OFF」と指定するという順番にしてください。

最近のビデオカードはほとんどがハードウェアスクロールをサポートしていますので、このオプションを利用することは少ないでしょうが、あの有名なET4000など、少々前のビデオカードを利用するときはお世話になるかもしれません。

キーボード関係のドライバ

DOS/Vではさまざまなタイプのキーボードに対応するため、キーボードのタイプを指定することができるようになっています。

PC-DOSではKEYB.COMをAUTOEXEC.BATで組み込みますが、MS-DOSでは**キーボードドライバ**をCONFIG.SYSで組み込みます。

● KEYB.COM (PC-DOS)

形式：KEYB.COM キーボードコード、コードページ、キーボード定義ファイル

目的：キーボードのタイプを指定する

日本語でDOS/Vを利用する場合は、JIS準拠のキーボード配列が便利ですので次のように指定しておけば間違いありません。

```
KEYB □ JP,932,C:¥DOS¥KEYBOARD.SYS ↵
```

英文タイプの熟練者は英語キーボードの方が打ちやすいかもしれません。その場合は次のようになります。

```
KEYB □ US,437,C:¥DOS¥KEYBOARD.SYS ↵
```

また、オプションをつけずに「KEYB↵」と実行すると、現在のキーボードの設定が表示されます（画面45）。

◎画面 45　現在のキーボードコードの設定

```
C:¥>KEYB
現在のキーボード・コード: JP  コード・ページ: 932
CON 装置では活動コード・ページが使用可能になっていません

C:¥>_




英数 半角  R
```

この KEYB.COM は常駐型プログラムですから、AUTOEXEC.BAT で組み込んでも、CONFIG.SYS の INSTALL コマンドで組み込んでもいいのですが、通常は AUTOEXEC.BAT の方で組み込まれます。

● JKEYB.SYS（MS-DOS）

形式：JKEYB.SYS キーボードタイプ キーボード定義ファイル
目的：キーボードのタイプを指定する

キーボードタイプとして指定できるのは次の通りです。

■ 106

一般に使用されている DOS/V 用のキーボードは「106キーボード」と呼ばれています。これはキーが106個あるからですが、通常はこの106を指定します。

■ 101

キーの数が少ない（101個）英語用のキーボードを使用する場合は、こちらを指定してください。なかなか小振りで便利ですがやはり日本語入力には

向きません。

AX

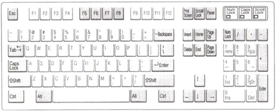
AX パソコンで DOS/V を利用する時にはこのオプションとなります。

DOS/V を106キーボードで使用するという設定は次のようになります。

```
DEVICEHIGH = C:¥DOS¥JKEYB.SYS □/106□C:
¥DOS¥JKEYBRD.SYS ↵
```

DOS/V パソコンに付属しているキーボードはほとんどが106ですので、このままの設定で大丈夫ですが、たまには101もあります。自分のキーボードがどちらかわからないかたはキーの数を数えて下さい。それも面倒という方のために101キーボードの図を載せておきましょう。

◎図　101キーボード（英語版）



画面を制御するドライバ

DOS/V では画面制御を行う「**エスケープシーケンス**」を利用することができます。これを利用すれば、文字に色をつけたり好きな位置にカーソルを移動したりといったことができますが、このエスケープシーケンスを利用するには「ANSI.SYS」というドライバを組み込まなければなりません。

● ANSI.SYS

形式：ANSI.SYS

目的：エスケープシーケンスを有効にする

このドライバを組み込めばエスケープシーケンスを利用することができます。ただし、エスケープシーケンスは DOS 上でのみ有効で、Windows3.1 にはまったく関係ありませんから、Windows3.1 を中心に使うユーザーにとっては不要です。

また、DOS で作業をするユーザーでも、このドライバを外してしまっても影響がない場合も多いので、思い切って CONFIG.SYS から削除してしまってもいいでしょう。特定のソフトウェアを使用して画面が乱れるなどの不具合が出たら組み込めばいいだけです。

なお、PC-DOS には ANSI.SYS の日本語版「**$ANSI.SYS**」が付属しています。これは ANSI.SYS に日本特有の機能をつけ加えたものですから、ANSI.SYS とほぼ同等の扱い方でかまわないでしょう。

画面モードを変更する

DOS/V では、文字をグラフィックとして描画していますので、さまざまな大きさのフォントを利用することができます。つまり同じ[3]**VGA** だとしてもフォントを小さくすれば、１画面に表示できる情報量は多くなるという理屈です。

もちろん、すっぴんの DOS/V ではそんな芸当はできません。何かドライバが必要です。

● V-Text

PC-DOS には **「V-Text」** ドライバが附属しています。このドライバを利用すれば、ちょっとだけ画面に表情を持たせることができます。

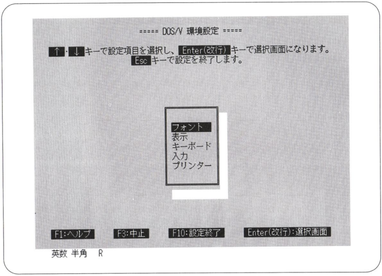
この「V-Text」ドライバを利用するためには、若干の設定を行わなければなりません。次の手順で設定を行いましょう。

1．SETUPV を起動する

「SETUPV ⏎」と入力して DOS/V 環境設定プログラムを起動します(画面46)。

[3]640×480の解像度のグラフィックモード DOS/V ではこの VGA が標準。

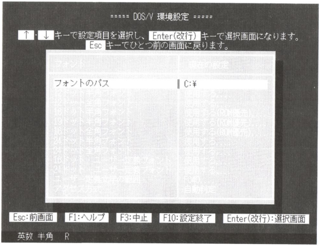
◎画面46　SETUPV 起動画面



2．フォントの設定を行う

「フォント」を反転させた状態で↵キーを押すと、フォントの詳細を設定する画面となります（画面47)。

◎画面 47　フォントの詳細を設定する画面

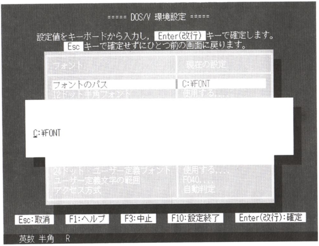


フォントのパス

フォントがどこに格納されているかを指定します。デフォルトではCドライブのルートディレクトリになっていますが、実際にフォントが格納されているディレクトリを指定してください。

指定方法はまず「フォントのパス」の行が反転されている状態で⏎キーを押し、表示されるウィンドウにフォントのパスを入力します（画面48）。入力したら⏎キーを押してください。

◎画面 48　フォントのパスを変更する

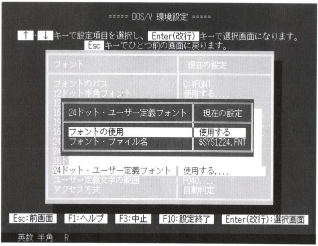


フォントの設定を変更する

12ドットから24ドットまでのフォントのすべてを「使用する」に設定してください。

「使用しない」となっている行を反転させ⏎キーを２回押すと、そのフォントを使用するかしないかを選択することができます（画面49)。「使用する」を反転させ⏎キーを押してください。

◎画面 49　フォントを使用するかしないかの選択

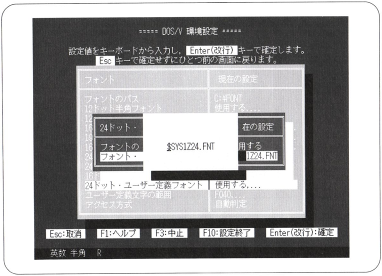


フォント・ファイル名を確認する

フォントを「使用する」に設定すると、画面49-1のようにフォントファイル名も設定できるようになります。通常は特に変更する必要はありませんが、24ドット全角フォント「**$JPNZN24.IBM**」だけはここでフォントファイル名を変更しなければなりません。

「フォント・ファイル名」を反転させ↵キーを押してください。フォントファイル名を指定できるウィンドウが表示されますから、「**$JPNZN24.FNT**」と変更しましょう。すでにこの名前になっていれば変更の必要はありません。変更したらやはり↵キーを押してください。

◎画面 49-1　フォント・ファイル名を変更する

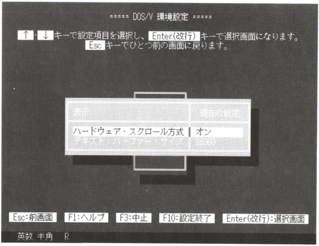


フォントをすべて「使用する」に設定したら、 ESC キーを押して最初の画面に戻ってください。

3．表示の設定を行う

「表示」を反転させ⏎キーを押すと、画面表示に関するオプションを指定することができるようになります（画面50）。

◎画面50　画面表示のオプションを指定する



ハードウェア・スクロール方式

「自動設定」にしておけば間違いはありませんが、とりあえず「オン」に変更してみてください。

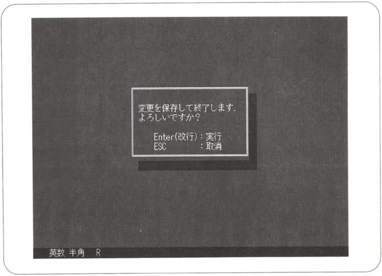
テキスト・バッファー・サイズ

テキストバッファとして使用するメモリサイズをバイト単位で指定してください。さまざまな画面表示モードを楽しむには、ここに「16,960」を指定しなければなりません。これは「最大行数×桁数×4」という計算になりますが、V-Textでは最大行数=40、最大桁数=106となりますからこの「16,960」が導き出されます。

4．設定を保存する

ここまでの設定を行ったら F10 キーを押してください。終了の方法についてのメッセージが表示されます（画面51）。↵キーを押すとこれまでの設定を保存して終了します。

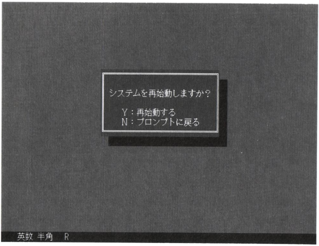
◎画面５１　終了の方法についてのメッセージ



５．システムを再起動する

続いてシステムを再起動するかどうかのメニューが表示されます（画面52)。ここでの設定はDOSV.INIファイルに書き込まれますが、その設定を有効にするにはマシンを再起動しなければなりません。［Y］を入力してパソコンを再起動しましょう。

◎画面 52　システムを再起動するかどうかの選択



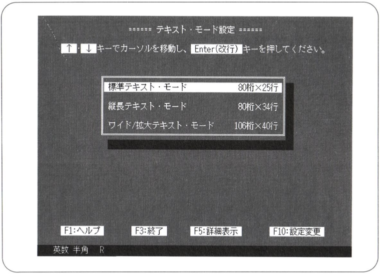
6．画面モードを設定する

パソコンが再起動されたら画面モードを変更してみましょう。まず次の2行を実行してください。

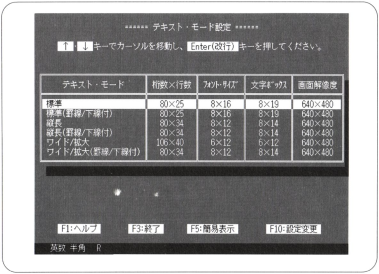
```
DSPXVGA ↵
DSPX ↵
```

これで画面モードを選択することができるようになります（画面53）。もっと詳しく見たい場合は **F5** キーを押してください（画面54）

◎画面 53　テキストモードを選択する



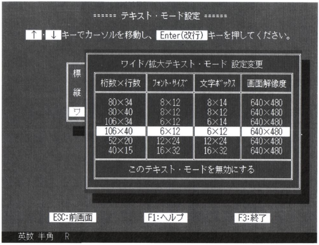
◎画面 54　詳しいテキストモード選択画面



ここに表示される画面モードがすべてではありません。変更したいテキストモード（標準以外）を反転させ **F10** キーを押すと、さらに数種のテキ

ストモードを選択できます（画面55)。

◎画面 55　テキストモードを選択する



VGA で選択できるテキストモードは表３のようになります。自分の好みに合ったテキストモードを指定してください。

◎表３　VGA で選択できるテキストモード

テキストモード	桁数×行数	フォント・サイズ	文字ボックス	画面解像度
標準	85×25	80×16	8×19	640×480
標準（罫線/下線付）	80×25	80×16	8×19	640×480
縦長	80×34	80×12	8×14	640×480
縦長（罫線/下線付）	80×34	80×12	8×14	640×480
ワイド/拡大	106×40	80×12	8×12	640×480
ワイド/拡大 (罫線/下線付)	80×34	80×12	8×14	640×480

7．AUTOEXEC.BAT に書き込む

このようにいちいちパソコンを起動するたびにテキストモードを指定するのも面倒です。AUTOEXEC.BAT に書き込んでおきましょう。

まず**「DSPXVGA」**はそのまま LH コマンドで組み込みます。

```
LH □ DSPXVGA ↵
```

「DSPX」の方ですが、これには選択するテキストモードに対応するオプションをつけて記述します。オプションは画面56のように５つです。これで起動と同時に指定したビデオモードで画面表示を行うことができます。

◎画面 56　DSPX のオプション

```
C:¥>DSPX /?
テキスト・モードの変更、および各テキスト・モードの設定変更を行います。

DSPX [ テキスト・モード [/EXT] ]

  テキスト・モード
    S   標準テキスト・モード
    L   縦長テキスト・モード
    W   ワイド/拡大テキスト・モード

  /EXT  罫線/下線付きテキスト・モード

DSPX D

  80桁×25行、16ドットフォントを使用する基本（従来）テキスト・モード
  このパラメーターは、文字などが正しく表示されない場合にのみお使いください。

オプションを指定しないで実行すると、メニュー形式で選択できます。

C:¥>_


    英数 半角   R
```

この画面モードはいつでも簡単に変更することができます。「DSPX ↵」でテキストモード設定を変更すればいいのです。

これでさまざまな画面モードを楽しむことはできますが、基本となる画面モードはあくまでも640×480の VGA モードです。ところが、最近の DOS/V マシンに搭載されているビデオカードは、ほぼすべてが1024×768程度の**高解像度**（ハイレゾリューションモード）を表示することができるようになっていますから、これではちょっと物足りなく感じるかもしれません。さらにその高解像度を DOS/V で楽しむなら、別途ドライバを手に入れなければなりません。

現在市販されている高解像度を実現するソフトウェアは、**「DOS/V エクステンション」**（IBM 製）と**「スーパードライバーズ32」**です。現在は、ほとんどが Windows3.1 環境に移行しつつありますので、DOS 画面を高解像度で表示させる V-Text はあまり需要がないかもしれません。興味のある方は購入してインストールしてみてください。

周辺機器を制御するドライバ

DOS/V マシンのハードウェアは PC/AT 互換機そのものです。この PC/AT はオープンなアーキテクチュアが特徴です。つまりデバイスドライバを利用しさえすれば、物理的なギャップを吸収できるので、本体と周辺機器との相性の問題や接続のトラブルがほとんどありません。

このように、DOS/V マシンではデバイスドライバの重要性が非常に高く、周辺機器を増設する時には欠かせないものとなっています。

●プリンタドライバ

プリンタドライバは、DOS から直接プリントアウトするためのものです。したがって、アプリケーションからプリントアウトする場合は、プリンタドライバは必要ありません。各アプリケーションがプリンタドライバを内蔵していて、そのアプリケーションからプリンタを直接制御してくれます。

DOS から直接プリントアウトする場合というのは、たとえば、TYPE コマンドでテキストファイルをプリンタへ**リダイレクト**するといった場合のことです。この説明で合点のいくかたはプリンタドライバを利用する機会もあるでしょうが、まったくわけの分からない方はプリンタドライバを組み込む必要はありません。

■ PC-DOS のプリンタドライバ

PC-DOS には、EPSON の[4]**ESC/P モード対応**のものと、IBM 5575/5577系のプリンタ対応のもの、さらに英語プリンタを ESC/P や5575/5577モードにエミュレートして漢字をプリントできるようにするドライバが附属していますが（表４）、これらはプリンタ機種に依存する常駐部分で、本来のデバイスドライバ部分「**$PRN.SYS**」は別になっています。

[4]エプソンが提唱するプリンタ制御コマンド規格。

◎表４　PC-DOS のプリンタ別常駐部分

ドライバ名	機　能
PRNESCP.COM	日本語 ESC/P プリンタドライバ
PRNIBM.COM	IBM PS/55 系日本語プリンタドライバ
EPCLESCP.COM	英語 PCL プリンタ上で ECS/P をエミューレト
EPCLIBM.COM	英語 PCL プリンタ上で5575/5577系プリンタをエミューレト

このようにデバイスドライバと常駐部分を分離することで、未使用時にはメモリ消費を押さえ、使用時にも自由にプリンタを切り替えることができるようになりました。

デバイスドライバ部分は CONFIG.SYS で組み込まなければなりません。

```
DEVICEHIGH = C:¥DOS¥$PRN.SYS ↵
```

一方常駐部分は CONFIG.SYS の INSTALL コマンドで組み込むこともできますが、その性格から考えると、AUTOEXEC.BAT で組み込むか起動後任意の時にコマンドラインから実行して組み込む方がいいでしょう。そうしておけば、/R スイッチで常駐を解除することもできます。たとえば、ESC/P 対応のプリンタを使用する場合は起動後、「**PRNESCP ↵**」とするだけで OK ですし、取り外すときは「**PRNESCP □/R ↵**」です。

■ MS-DOS のプリンタドライバ

MS-DOS では「**JPRINTER.SYS**」という ESC/P 対応のプリンタドライバを使用します。このドライバは CONFIG.SYS で組み込みます。

```
DEVICEHIGH = C:¥DOS¥JPRINTER.SYS ↵
```

なお、PC-DOSの常駐部分のプログラムにしろ、MS-DOSのJPRINTER.SYSにしろ、若干のオプションを指定することができますが、ここでは省略させていただきます。

また、何度も言うようですが、これらのプリンタドライバは通常のアプリケーションからの印刷にはまったく不要のものですから、できるだけ組み込まないようにしておきましょう。Windows3.1でもWindows3.1専用のプリンタドライバを使いますので、やはり不要です。

● FDDドライバ

もともと3.5インチフロッピーディスクは1.44Mバイトの容量でしたが、最近では1024バイト/セクタまたは512バイト/セクタの1.2Mバイトも読み書きできる**3モードドライブ**が主流です。

とはいえ、もともとサポートしていないタイプのフォーマットを読み書きするのですから、そのためのデバイスドライバが必要です。

$FDD12.SYS (PC-DOS)

形式：$FDD12.SYS

目的：1.2MのFDDを読み書きできるようにする

PC-DOSではこのドライバで1.2Mバイト（1024バイト/セクタ、512バイト/セクタ）のフロッピーディスクへアクセスすることができるようになります。

組み込みはCONFIG.SYSで次のように記述して行います。

```
DEVICEHIGH = C:¥DOS¥$FDD12.SYS ↵
```

このデバイスドライバが組み込まれるとメッセージが表示されますので、確認してください（画面57）。

◎画面 57 組み込まれたメッセージを確認する

```
PC DOS Power Manager Version 1.0
Copyright (c) IBM Corporation 1986, 1991-1993

DOS/V プリンター・ドライバ・マネージャー(V1.01)が導入されました。
3.5 型ディスケット駆動機構デバイス・ドライバー($FDD12.SYS Ver 2.02)
(C) Copyright IBM Corporation 1991,1994. All rights reserved.
3.5 型ディスケット駆動機構デバイス・ドライバーが導入されました。
IBM RAMDrive version 3.06 virtual disk E:
    ディスク・サイズ        : 2048K
    セクター・サイズ        : 256 バイト
    クラスター・サイズ      : 4 セクター
    ディレクトリー項目数    : 128
連文節変換ルーチン (IBMMKKV.EXE Version 6.03)が EMS メモリーに導入されました。
IBM PS/55系日本語プリンター・ドライバー(PRINIBM.COM Ver. 2.00)が導入されました。


PC DOS Version J6.3/V

Revision 0


C:¥>_
    英数 半角  R
```

組み込まれたメッセージ

■ 1024FD.EXE (MS-DOS)

形式：1024FD.EXE オプション

目的：1.2M の FDD を読み書きできるようにする

PC-DOS の$FDD12.SYS の MS-DOS 版です。こちらはデバイスドライバでなく常駐型プログラムとなっています。組み込みは CONFIG.SYS で行われますが、オプションを利用することによって UMB を利用しない設定を行うことができます。

DEVICEHIGH = C:¥DOS¥1024FD.EXE ↵
DEVICEHIGH = C:¥DOS¥1024FD.EXE □/L ↵
UMB を使わない場合

また、1024FD.EXE をメモリから解放することができます。こちらはもちろんコマンドラインから実行します。

1024FD.EXE □/R ↵

1.2 バイトのフロッピーディスクへアクセスできるようにした場合、それ

らディスクのフォーマットも行うことができますが、DOSのFORMATコマンドではなく、特別なフォーマットプログラムを使用するようになっています。

アクセスする場合のドライブ名は機種によって、またドライブによって違います。Thinkpad230CsではAドライブBドライブのどちらでもアクセスできるようになりますが、DやEドライブを使用するものもあります。また、ハードウェアスイッチでアクセスするメディアサイズを変更しなければならないものもあります。

なお、1.2Mバイトのフロッピーディスクにアクセスできるのは、あくまでもハードウェア的に1.2Mバイトをサポートしているフロッピーディスクドライブを使用している場合です。

２モードのフロッピーディスクでは、いくらデバイスドライバを組み込んでも1.2Mバイトのフロッピーディスクを読み書きすることはできません。

なお、DOS/V附属の普通のFORMATコマンドでは、1.2Mバイトのフォーマットはできません。**FORMAT12.COM**（PC-DOSもMS-DOSも）を使ってください（画面58）。

◎画面58　FORMAT12.COM

```
C:¥>FORMAT12
1.2MBディスケット用フォーマット・ユーティリティー
バージョン 2.02
(C) Copyright IBM Corporation 1991, 1994


 (1)     1．2MBフォーマット（1024バイト／セクター）
 (2)     1．2MBフォーマット（ 512バイト／セクター）

フォーマット・タイプを番号で入力してください ===> 2

ドライブを指定してください (A,B) ===============> A

新しいディスケットをドライブ A: に入れてください
準備ができたら ENTERキーを押してください ..._




    英数 半角   R
```

●マウスドライバ

DOSアプリケーションでマウスを利用するには**マウスドライバ**を組み込まなければなりませんが、マウスドライバは「MOUSE.COM」という常駐型プログラムとなっています。

常駐させる

常駐型ですから、AUTOEXEC.BATで組み込むことも、CONFIG.SYSのINSTALLコマンドで組み込むこともできますが、やはり性格から考えるとAUTOEXEC.BATで組み込むのが正解です。

ただし、Windows3.1を利用する上では、このマウスドライバは関係ありませんから、たとえば、DOSSHELLのように、マウスを使用するDOSアプリケーションを利用しないユーザーは不要なドライバです。

また、マウスを利用するDOSアプリケーションを使う場合も、そのアプリケーションを起動するバッチファイルを作って、そこに「MOUSE」と一行書いておけばいいので、AUTOEXEC.BATで組み込む必要もありません。

オプション

「MOUSE.COM」には若干のオプションがありますが、通常起動時にはなにも指定する必要はありません。ただし、非常に繊細な作業が必要であるとか、もっと大胆にマウスを使用したいといった場合は、オプションでマウスの動作環境を設定することができます（画面59）。

◎画面 59　PC-DOS の MOUSE.COM

```
C:¥>MOUSE /?
Microsoft (R) Mouse Driver Version 8.20
Copyright (C) Microsoft Corp. 1983-1992.
Copyright (C) IBM Corp. 1992-1994.

マウスドライバーをインストールします。

MOUSE [マウス タイプ] [割り込みレート] [マウス感度]
      [加速度] [カーソル] [ハードウェア カーソル] OFF

マウス タイプ    /Cn (シリアル マウス), /Z (PS2 マウス),
                 /In (インポート マウス), /B (バスマウス)
                  (nはポート番号で 1または2を指定します)
割り込みレート   /Rn (nは割り込みレートで 0～4の値を指定します)
マウス感度       /Hn (水平方向), /Vn (垂直方向), /Sn (水平・垂直方向)
                  (nには0～100の値を指定します)
加速度           /Pn (nには加速度プロファイルの番号を指定します)
カーソル         /Nn (カーソル動作の緩慢さ), /M (デフォルト カーソル)
                  (nは緩慢動作の遅延値で 0～255を指定します)
ハードウェア カーソル
                 /Y (ハードウェア カーソルを使用します)
OFF              マウスドライバーをメモリーから削除します

C:¥>_
    英数 半角   R
```

常駐解除

「MOUSE.COM」の常駐を解除するオプションは「OFF」です。PC-DOS でも MS-DOS でも次のように指定します。

MOUSE ␣ OFF ↵

かな漢字変換機能のためのドライバ

DOS/V で日本語を入力するには、かな漢字変換機能を利用できるようにしなければなりません。

● IBMMKKV.EXE（PC-DOS）

形式：IBMMKKV.EXE オプション

目的：連文節かな漢字変換機能を組み込む

PC-DOS には **IBMMKKV.EXE** という連文節かな漢字変換機能プログラムが付属しています。PC-DOS がプリインストールされている機種では、すでにこのかな漢字変換機能が利用できるようになっています。また、PC-DOS をインストールする時には、かな漢字変換機能を組み込むかどうか選択できますので、そこで選択すれば PC-DOS のインストールと同時にかな漢字変換機能を組み込むことができます。

IBMMKKV.EXE は常駐型プログラムですので、CONFIG.SYS の INSTALL コマンドで組み込みますが、その主なオプションは次のようなものがあります。

■ /C：自動文節区切り直しをする

■ /I：IBM 漢字コードを使用

■ /J＝78｜83｜90　：JIS 区点コードセットを使用する

78：JIS78 を使用する
83：JIS83 を使用する
90：JIS90 を使用する

■ /K：文字確定を⏎キーに割り当てる

■ /L：辞書学習結果を保存する

■ /M=B|F|I|J|S　：変換機能の選択

B：単文節変換
F：複合変換
I：連文節一括変換
J：自動変換
S：連文節先読み変換（デフォルト）

■ /R：RAM辞書を使用する

■ /S=プロファイルファイル名　：辞書プロファイル名の指定

■ /T：定位置変換を指定する

■ /U=辞書ファイル名　：ユーザー辞書の指定

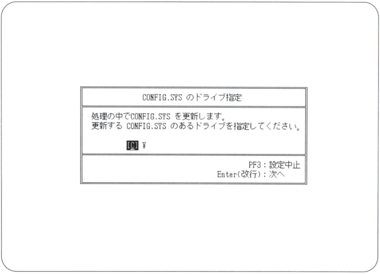
■ /Z=自動全候補の回数　：自動全候補の回数を指定

こんなにあるオプションをすべて使用する必要もなく、必要なオプションをつけておけばいいでしょう。ThinkPad230CsのCONFIG.SYSでは次のように指定されています。

```
INSTALL=C:¥DOS¥IBMMKKV.EXE□/M=S□/Z=4□/C□/
L□/J=90□/S=C:¥DOS¥MULTDICT.PRO□/U=C:¥$U↵
```

DOS/VのインストールにIBMMKKV.EXEを組み込まなかった場合でも、上記のようにCONFIG.SYSに記述してかな漢字変換機能を組み込むことができます。また、IBMMKKV.EXEの各種設定は「SETUPM-KK.EXE」というプログラムで行うことができます（画面60）。

◎画面60　SETUPMKK.EXE



なお、PC-DOS には「$IAESKK.SYS」という漢字を一文字ずつ変換する単漢字変換機能のかな漢字変換プログラムも附属していますが、現在単漢字変換を使用するメリットはほとんどありません。

● MSIME.SYS（MS-DOS）

形式：MSIME.SYS オプション

MSIME.SYS はデバイスドライバですから、CONFIG.SYS の DEVICE コマンドで組み込みます。ただし、MSIME を組み込むには、その前に MSIMEK.SYS というプログラムを組み込んでおかなければなりません。つまり次のようになります。

```
DEVICE = C:¥DOS¥MSIMEK.SYS □/A1
DEVICE = C:¥DOS¥MSIME.SYS □/D*C:¥DOS¥MSIMER.DIC
□/DC:¥DOS¥MSIME.DIC □/C1 □/N □/A1 ↵
```

オプションは非常に多岐にわたりますので、ここでは重要なもののみをご紹介します。

■ /An　：拡張メモリの使用方法を指定する

n = 1：EMS メモリを使用する
　　2：XMS メモリを使用する
　　3：EMS、XMS の両方を使用する

■ /Cn　：使用する日本語のコード体系

n = 0：JIS
　　1：シフト JIS
　　2：区点

■ /D ＊ファイル名のパス：システム辞書のパスを指定する

■ /Hnm：変換方式と自動確定を指定する

n = 0：連文節変換
　　1：複合語優先変換
　　2：自動変換
　　3：無変換
m = 0：自動確定する
　　1：自動確定しない

■ /L　：学習機能をオフにする

■ /N　：エコーラインに入力する

■ /Y　：ゆらぎ処理

MS-DOS をインストールするときに、やはりかな漢字変換プログラムを組み込むかどうかの選択を行うことができます。

MS-DOS に付属の MSIME は次のように CONFIG.SYS で組み込まれます。

```
DEVICEHIGH = C:¥DOS¥MSIMEK.SYS □/A1 ↵
DEVICEHIGH = C : ¥ DOS ¥ MSIME. SYS □/D ＊ □ C : ¥ DOS
¥MSIMER.DIC □/DC:¥DOS¥MSIME.DIC □/C1 □/N □/A1 ↵
```

このMSIMEの設定を変更するには、MSIMESET.EXEというプログラムを使用します（画面61）。

◎画面61　MSIMESET.EXE

```
ＭＳ ＩＭＥ環境設定　1/2                                        Ver 2.53
─────────────────────────────────────────────
                                    ↑↓←→：選択　Enter：決定　Esc：終了
入力ファイル [C:¥CONFIG.SYS                                          ]
出力ファイル [C:¥CONFIG.SYS                                          ]
システム辞書 [C:¥MSIMER.DIC                                          ]
ユーザー辞書 [C:¥MSIME.DIC                                           ]
環境ファイル [                                                       ]
システムパス [C:¥                                                    ]
展開バッファ [  10] ＫＢ　(5～32)
辞書バッファ [   7] ＫＢ　(6～32)
入力方式　　 [ローマ]　[かな]
文字幅　　　 [全角]　　[半角]
変換方式　　 [連文節]　[複合語優先]　[自動]　[無変換]
辞書学習　　 [する]　　[しない]
コード　　　 [ＪＩＳ]　[シフトＪＩＳ] [ＪＩＳ区点]
句読点　　　 [、。]　　[，．]　　[、．]　　[，。]
記号　　　　 [「」・]　[［］／]　[「」／]　[［］・]
入力位置　　 [ｼｽﾃﾑﾗｲﾝ] [エコー]



[一　覧][　　][　　][次画面][　　][実　行][　　][終　了]
```

●$IAS.SYS（PC-DOS）

形式：$IAS.SYS オプション

目的：かな漢字変換機能のサポート

かな漢字変換プログラムをサポートするドライバです。正常に組み込まれると画面左下に「英数 半角　R」というように表示されます。

オプションは次の通りです。

■ /G=n

n=0：システムライン（画面の一番下）で変換（デフォルト）

n＝1：エコーライン（カーソルの位置）で変換

/R＝n

n＝0：ローマ字モードオフ（デフォルト）
n＝1：ローマ字モードオン

/X＝n

n＝0：EMSメモリを使用しない
n＝1：EMSメモリを使用する（デフォルト）

/K＝JP｜US｜AX｜J3

JP：日本語106キーボードを使う
US：英語キーボードを使う
AX：AXキーボードを使う
J3　：J-3100ノートブック型のキーボードを使う

他のかな漢字変換機能を組み込む場合は、この$IAS.SYSはあってもなくてもかまいません。メモリ状況から考えると、とりあえず外しておいて不都合がでてきたらこの$IAS.SYS（またはKKCFUNK.SYS）を組み込みます。

●単漢字変換

PC-DOSには「$IAESKK.SYS」という単漢字変換プログラムが付属しています。この単漢字変換とは、一文字ずつ漢字に変換していくもので、連文節変換などのプログラムに比べると使い勝手の悪いことおびただしい代物です。

たしかにサイズが小さく辞書も小さいというメリットはあるものの、現在のように大容量のハードディスクとメモリを搭載しているマシンにとっては、そのメリットもいかほどのものかと思えてしまいます。この単漢字変換機能を利用することはお勧めできませんが、組み込み方くらいはご紹介しておきましょう。

```
DEVICEHIGH = C:¥DOS¥IAESKK.SYS ↵
```

なお、この単漢字変換の方は、必ず$IAS.SYSを組み込み、その後に組み込まなければなりません。

●他のかな漢字変換プログラム

DOSに付属のかな漢字変換プログラムを使用するのが一番簡単ですが、ATOK9などのかな漢字変換プログラムを使用することもできます。この場合、インストール方法は2つあります。

かな漢字変換プログラムだけを購入した場合

かな漢字変換プログラムの中には単体で販売されているモノもあります。そのプログラムを購入しインストールすれば、DOS付属のかな漢字変換プログラムのかわりに使用することができます。

アプリケーションプログラムに付属のものを使う

アプリケーションプログラムにかな漢字変換プログラムが付属している場合があります。とくにワープロソフトには、必ずといっていいほどかな漢字変換プログラムが付属しています。そのソフトをインストールすれば自動的にかな漢字変換プログラムもインストールされることがあります。

たとえば、ワープロのベストセラー一太郎Ver.5にはATOK8というかな漢字変換プログラムが付属していますが、一太郎Ver.5をインストールすれば自動的にATOK8もインストールされるようになっていますし、一太郎Ver.5はこのかな漢字変換プログラムがないと動きません。

どちらにしろ、かな漢字変換プログラムをインストールするインストーラーが付属していますので、ユーザーは対話式でオプションを選択していくことにより簡単にCONFIG.SYSで組み込むことができます。

たとえば、ThinkPad230Csのデフォルトの環境にATOK8を組み込むと、次のようなCONFIG.SYSになります。もちろんもともとのIBMM-KKV.EXEは削除しておきます。

```
BUFFERS = 20
FILES = 30
DOS = HIGH,UMB
COUNTRY = 081,932,C:¥DOS¥COUNTRY.SYS
SHELL = C:¥DOS¥COMMAND.COM □/P □/E:512
DEVICE = C:¥DOS¥$FONT.SYS
DEVICE = C:¥DOS¥HIMEM.SYS
DEVICE = C:¥DOS¥EMM386.EXE RAM
DEVICEHIGH = C:¥DOS¥SETVER.EXE
DEVICEHIGH = C:¥DOS¥$DISP.SYS
DEVICEHIGH = C:¥DOS¥$IAS.SYS
DEVICE = C:¥EZGOING¥IBMDSS01.SYS /M
DEVICE = C:¥EZGOING¥IBMDOSCS.SYS
DEVICE = C:¥EZGOING¥DICRMU01.SYS
DEVICE = C:¥EZGOING¥$ICPMDOS.SYS
DEVICE = C:¥DOS¥POWER.EXE
DEVICE = C:¥EZGOING¥AUTODRV.SYS □ C:¥EZGOING¥AUTODRV.INI
DEVICE = C:¥EZGOING¥DISKDRV.SYS
DEVICEHIGH = C:¥DOS¥$PRN.SYS
DEVICEHIGH = C:¥DOS¥ANSI.SYS /X
DEVICEHIGH = C:¥DOS¥$FDD12.SYS
REM DEVICEHIGH = C:¥DOS¥RAMDRIVE.SYS
STACKS = 9,256
LASTDRIVE = Z
DEVICE = C:¥ATOK¥ATOK8A.SYS □/UCF = C:¥ATOK¥ATOK8.UCF
DEVICE = C:¥ATOK¥ATOK8B.SYS
DEVICE = C:¥ATOK¥ATOK8IAE.SYS
```

ディスクキャッシュプログラムを組み込む

BUFFERS コマンドでは、ディスクから読み込んだデータを一時保管する（キャッシュする）ためのメモリの量を指定していますが、より強力な機能を持った専用のプログラム**「キャッシュ」**プログラムがあります。

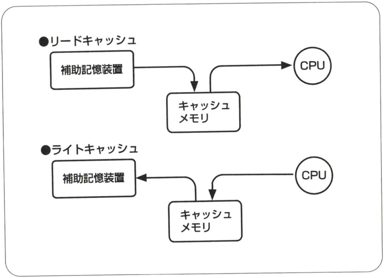
現在のパソコンではCPUの処理速度に比べ、どうしてもディスクの読み書き速度が落ちますので、全体のスループットを向上させるためにも、補助記憶装置からデータを読み出したり書き込んだりする場合に有効なキャッシュプログラムが欠かせません。

DOS/V マシンには通常「SMARTDRV」というキャッシュプログラムが附属しています。このSMARTDRVは原則としてAUTOEXEC.BATで組み込まれますが、ほとんどの場合デフォルトのAUTOEXEC.BATに記述されていますので、ユーザーがわざわざ組み込む必要はありませんが、その設定方法や使い方を知らないとひどい目に遭わないとも限りません。

●リードキャッシュとライトキャッシュ

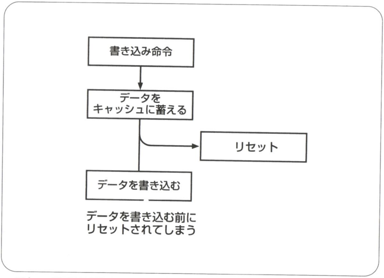
キャッシュには読み出し時にデータをキャッシュする**「リードキャッシュ」**と書き込み時にキャッシュする**「ライトキャッシュ」**とがあります（図16)。

◎図１６　リードキャッシュとライトキャッシュ



リードキャッシュはほぼすべての場合に有効ですが、ライトキャッシュは気をつけなければならない面が多い機能です。というのも、ライトキャッシュはデータの保存を実行した時と、データがディスクに記録される時にタイムラグが生じ、その間にパソコンをリセットしたりしてしまうとせっかく書き込んだと思ったデータが書き込まれなかったりします（図17）。

◎図 17　書き込む前にリセットしてしまう



現在の SMARTDRV では、特に指定しないかぎりライトキャッシュは行われない設定ですが、もしも SMARTDRV でライトキャッシュを設定していたり、他のキャッシュプログラムを利用している場合は気をつけてください。

とくにリブートするコマンドを利用するバッチプログラムでは、このトラブルが起こることが多いので、キャッシュ内容を強制的にディスクに書き出すコマンドを記述しておきましょう。たとえば SMARTDRV なら「/C」スイッチをつけて「SMARTDRV □/C」という１行を加えておきます。

●キャッシュの設定を確認する

適切なキャッシュを設定するにしろ、現在の状況がわからなければしょうがありません。たとえば、SMARTDRV なら何もオプションをつけずに「**SMARTDRV** ↵」とすれば現在の状況が表示されます（画面62）。

◎画面 62　SMARTDRV の設定を確認する

```
C:¥>SMARTDRV
IBM SMARTDrive Disk Cache Version 5.0
Copyright (c) IBM Corp. 1991, 1994
キャッシュ・サイズ: 2,097,152 バイト
ウィンドウズ実行時のキャッシュ・サイズ: 2,097,152 バイト

              ディスクキャッシュ情報
ドライブ  リード・キャッシュ  ライト・キャッシュ  バッファリング
----------------------------------------------------
  A:        yes           no            no
  B:        yes           no            no
  C:        yes           no            no
  D:        yes           no            no

キャッシュされたデータはDOSプロンプトに戻る前に書き込まれます。

ヘルプを見るには、"smartdrv /?"と入力して下さい

C:¥>_

     英数 半角   R
```

キャッシュサイズ

SMARTDRV のために確保されるサイズです。

ウィンドウズ実行時のキャッシュサイズ

ウィンドウズを実行する時に確保される SMARTDRV のためのメモリサイズです。

ディスクキャッシュ情報

ここはデフォルトではリードキャッシュがすべて yes（または、「する」）になっています。

ライトキャッシュは PC-DOS と MS-DOS でデフォルトの設定が違い、PC-DOS ではすべて no であるのに対し、MS-DOS では、ハードディスクについては on でその他のドライブ（CD-ROM やフロッピーディスク）については off です。

バッファリング

ここも通常はすべて no です。めったにないことですが、もしも yes (または「する」「必要」) になっている場合は、CONFIG.SYS に次の1行があることを確認してください。もしなければ追加しなければなりません。

```
DEVICE = C:¥DOS¥SMARTDRV.EXE/DOUBLE_BUFFER↵
```

● SMARTDRV のオプション

キャッシュするディスクやそのオプションの設定はキャッシュプログラムによって違いますが、SMARTDRV の場合は画面63のようになっています。

◎画面 63 SMARTDRV のオプション

```
SMARTDRV [/X] [[drive[+|-]]...] [/U] [/C | /R] [/F | /N] [/L] [/V | /Q | /S]
         [InitCacheSize [WinCacheSize]] [/E:ElementSize] [/B:BufferSize]

/X              全ドライブに対して書き込みキャッシュを禁止します
ドライブ文字     キャッシュを行うドライブを指定します。
                ドライブ文字のみの場合は読み出しのみキャッシュを行います
+               指定ドライブに対して書き込みにもキャッシュを行います
-               指定ドライブに対してキャッシュを行いません
/U              CD-ROM キャッシュモジュールをロードしません
/C              キャッシュ内に書きためた全データをハードディスクに書き込みます
/R              キャッシュ内の全データをクリアしてSMARTDriveを再スタートします
/F              プログラム終了時にキャッシュしたデータを書き込みます (省略値)
/N              プログラム終了時にキャッシュしたデータを書き込みません
/L              基本メモリにSMARTDriveをロードします
/V              スタート時に SMARTDrive の動作状況を表示します
/Q              スタート時に SMARTDrive の動作状況を表示しません
/S              SMARTDriveの状態を示す追加情報を表示します。
InitCacheSize   キャッシュが使用するXMSメモリ容量(KB)を指定します
WinCacheSize    ウィンドウズ実行時に使用するXMSメモリ容量(KB)を指定します
/E:ElementSize  キャッシュ・エントリーのサイズをバイト単位で指定します
/B:BufferSize   読み出しバッファのサイズを指定します

C:¥>_
   英数 半角   R
```

/X

すべてのドライブに対してリードキャッシュを禁止します。PC-DOS では、このオプションをつけなくてもデフォルトではライトキャッシュは off ですので通常は必要ありませんが、MS-DOS ではハードディスクに対して

ライトキャッシュが on に設定されますので、その機能を無効にするためには AUTOEXEC.BAT で SMARTDRV を組み込む際にこのオプションを指定してください。

```
C:¥DOS¥SMARTDRV /X
```

drive

キャッシュを行うドライブを指定しますが、PC-DOS にしろ MS-DOS にしろ、デフォルトですべてのドライブに対してキャッシング（リードキャッシュ）が行われますので、次に解説する「-」オプションでキャッシュを無効にした場合、再度リードキャッシュのみを有効にすることができます。

たとえば、CとDドライブにリードキャッシュのみを設定するなら次のようになります。

```
C:¥DOS¥SMARTDRV C D↵
```

・+、「drive」で指定されたドライブはリードキャッシュのみ設定されます。ライトキャッシュも指定するには「+」オプションをつけなければなりません。

たとえば、CとDドライブにライトキャッシュを設定するには次のようにします。

```
SMARTDRV C+ D+
```

/U

CD-ROM キャッシュモジュールをロードしないオプションです。CD-ROM を使用しない場合はこのオプションを指定してもいいでしょうが、CD-ROM を使用しない場合は必ずこのオプションを指定するというものでもありません

/C

ライトキャッシュされたデータをすべてハードディスクに書き出します。キャッシュされたデータを強制的にディスクに書き込む際に使用します。し

たがって、リードキャッシュのみの設定ではこのオプションは意味がありません。

/R

キャッシュされたデータをすべてクリアして、SMARTDRVを再スタートさせます。

/F

プログラムの終了時に、ライトキャッシュデータを書き込みます。これはデフォルトのオプションですので、とくに指定する必要はありません。

/L

プログラムの終了時にキャッシュされたデータを書き込みません。これは恐ろしいオプションですので、よほどのことがない限り指定しない方がいいでしょう。

/V or /Q

スタート時にSMARTDRVの状況を表示するか（/V）しないか（/Q）を指定します。どちらでもお好きな方をどうぞ。

/S

より詳しい追加情報を表示することができます（画面64）。

◎画面 64　追加情報を付加した

```
C:¥>SMARTDRV /S
IBM SMARTDrive Disk Cache Version 5.0
Copyright (c) IBM Corp. 1991, 1994

   256 個のキャッシュ・エントリー (1エントリー   8,192 バイト) があります。
今までに、キャッシュ・ヒットが     630 回、
         キャッシュ・ミスが   2,785 回ありました。
キャッシュ・サイズ: 2,097,152 バイト
ウィンドウズ実行時のキャッシュ・サイズ: 2,097,152 バイト

              ディスクキャッシュ情報
ﾄﾞﾗｲﾌﾞ  ﾘｰﾄﾞ･ｷｬｯｼｭ     ﾗｲﾄ･ｷｬｯｼｭ     ﾊﾞｯﾌｧﾘﾝｸﾞ
----------------------------------------
  A:       yes           no           no
  B:       yes           no           no
  C:       yes           no           no
  D:       yes           no           no

キャッシュされたデータはＤＯＳプロンプトに戻る前に書き込まれます。


ヘルプを見るには、"smartdrv /?"と入力して下さい

C:¥>_
   英数 半角   R
```

InitCacheSize

SMARTDRV のために確保されるキャッシュサイズです。指定は K バイト単位です。

WinCacheSize

ウィンドウズはより多くのメモリを利用するために、ウィンドウズ起動時にここで指定したサイズまで使用メモリを圧縮します。

たとえば、通常のキャッシュサイズとして 4M バイト、ウィンドウズ用の最小キャッシュサイズとして 2M バイトを指定するなら、次のようになります。

```
C:¥DOS¥SMARTDRV□4096□2048↵
```

/E:ElementSize

キャッシュ・エントリーのサイズをバイト単位で指定します。このオプションは特に必要ではありません。

/B:BufferSize

読み出しバッファのサイズを指定します。このオプションも滅多に必要ではありません。

第4章
マルチ*CONFIG*で複数の環境を切り替える

DOS/Vでは「マルチCONFIG」という機能があります。これは、CONFIG.SYSにあらかじめ複数の環境を登録しておき、起動時に選択して環境を構築する機能です。

マルチCONFIGでは、CONFIG.SYSの内容をブロックごとに分け、そのブロックを選択するためのメニューという構成になります。

ここでは、ThinkPad230CsにプレインストールされているDOS/VのCONFIG.SYSとAUTOEXEC.BATをサンプルにマルチCONFIGの作り方を解説します。とりあえずそのCONFIG.SYS、AUTOEXEC.BATと各ブロックのご紹介をしておきましょう。

CONFIG.SYSとAUTOEXC.BAT

● ThinkPad230CSのCONFIG.SYSとAUTOEXEC.BAT

ThinkPad230CsのCONFIG.SYSとAUTOEXEC.BATは次のようになっていますが、CONFIG.SYSにはPCMCIAカード関係のドライバ類の記述がありますし、AUTOEXEC.BATの中には通常の使用にまったく不要なセクションもあるので、ここでは若干編集してあります。

CONFIG.SYS

```
BUFFERS = 20
FILES = 30
DOS = HIGH,UMB
```

```
COUNTRY = 081,932,C:¥DOS¥COUNTRY.SYS
SHELL = C:¥DOS¥COMMAND.COM □/P C:¥DOS □/E:512
DEVICE = C:¥DOS¥$FONT.SYS
DEVICE = C:¥DOS¥HIMEM.SYS
DEVICE = C:¥DOS¥EMM386.EXE □ RAM □ X = C800-CFFF
DEVICEHIGH = C:¥DOS¥$DISP.SYS
DEVICEHIGH = C:¥DOS¥$IAS.SYS
DEVICE = C:¥DOS¥POWER.EXE
DEVICEHIGH = C:¥DOS¥ANSI.SYS □/X
REM DEVICEHIGH = C:¥DOS¥RAMDRIVE.SYS
INSTALL = C:¥DOS¥IBMMKKV.EXE □/M = S □/Z = 4 □/C □/
L □/J = 90 □/S = C:¥DOS¥MULTDICT.PRO □/U = C:¥$USR-
DICT.DCT
STACKS = 9,256

-------------------------AUTOEXEC.BAT-------------------------
@ECHO □ OFF
PROMPT □$P$G
PATH □ C:¥WINDOWS;C:¥DOS;C:¥IBMVESA;C:¥EZGOING;%
PATH%;
SET □ TEMP = C:¥DOS
C:¥THINKPAD¥FUELDOS
C:¥DOS¥SMARTDRV.EXE
LH □ C:¥DOS¥KEYB.COM JP,932,C:¥DOS¥KEYBOARD.SYS
WIN
```

●ブロック

[MENU]

目的：メニューを作成し、実際に起動する環境（ブロック）を選択する

内容は次のようになります。

MENUITEM =ブロック名，メッセージ

ここに指定されたメッセージが起動時にメニューとして表示され、そのブロックが実行されます。

SUBMENU =サブメニュー名，メッセージ

「サブメニュー」で指定したメニューブロックの内容にしたがって、メニューが表示されます。

MENUDEFAULT =ブロック名，タイムアウト時間（秒単位）

メニューが表示されている間、「タイムアウト時間」で指定された時間内にユーザーがメニューの選択を行わなかった場合、「ブロック名」で指定されたブロックがデフォルトとして実行されます。

タイムアウト時間は0～99秒まで指定できますが、0を指定するとメニューを表示せずそのまま「ブロック名」で指定されたブロックが実行されます。

MENUCOLOR =文字色，背景色

メニューの文字色とその背景色を指定します。指定できる色は表6となっています。

◎表５　文字色と背景色

指定できる数値	表示色
0	黒
1	青
2	緑
3	シアン
4	赤
5	マゼンダ
6	茶
7	白
8	グレー
9	明るい青
10	明るい緑
11	明るいシアン
12	明るい赤
13	明るいマゼンダ
14	明るい黄
15	明るい白

ただし、パソコンの種類によっては８～１５を指定すると文字が点滅する場合があります。

[COMMON]

「COMMON」ブロックの内容は、メニューでどのブロックを選択されたかに関わらず実行されます。したがって、どの環境でも共通の環境は「COMMON」ブロックに記述しておきます。なお、COMMONブロックは１つのCONFIG.SYSに複数記述することが可能です。

[サブメニュー]

「MENU」ブロックの「SUBMENU」で指定されたブロックです。内容は「MENU」ブロックと同じです。

ブロック

実際の環境が記述されるブロックです。

●メニューを作る

まずメニューを作りましょう。起動する環境の分だけ「MENUITEM=」を記述していけばいいのです。ブロックやメッセージはそれぞれ半角70文字まで記述することができますが、まだ日本語のフォントがロードされていませんし、ディスプレイドライバも組み込まれていませんから、日本語を使ってもただ意味のない文字が表示されるだけです。

実例を見てみましょう。たとえば、DOS/Vの日本語モード、英語モード、Windows3.1環境というような3つの環境を切り替えるのであれば次のようになります。

```
[MENU]
MENUITEM = DOSV,Japanese ▯ Mode
MENUITEM = PC-DOS,English ▯ Mode
MENUITEM = WINDOWS,Windows3.1
MENUDEFAULT = WINDOWS,30
MENUCOLOR = 11,0
```

なお、一つのCONFIG.SYSで指定できるMENUITEMは9つまでです。10以上の環境を切り替える必要があるかどうかは別として、それを越えるようでしたら「SUBMENU」を使って下さい。

たとえば、日本語モードの中でさらにFEPの種類を指定するような場合、もちろん9つを越えなければMENUに記述した方が便利ですが、わかりやすいように階層を変えてSUBMENUに記述することもできます。

```
[MENU]
SUBMENU = DOSV,Japanese ▯ Mode
MENUITEM = PC-DOS,English ▯ Mode
MENUITEM = WINDOWS,Windows3.1
MENUDEFAULT = WINDOWS,30
MENUCOLOR = 11,0

[DOSV]
MENUITEM = ATOK8,ATOK8
```

```
MENUITEM = WX2,WX2
MENUDEFAULT = ATOK8,30
MENUCOLOR = 11,0
```

このような MENU 構造でマシンをリセットすると、まず「MENU」の内容が表示されそこで「DOS/V」を選択するとサブメニュー「DOS/V」の内容が表示されるという仕組みです。

サブメニューから一つ上の階層に戻るときは **BS** キーを押して下さい。

●[COMMON]

COMMON ブロックには、すべての環境に共通のものを記述します。たとえば、FILES、BUFFERS、DOS、SHELL などをここに記述します。また、「HIMEM.SYS」もすべての環境に共通ですから、ここに記述しておきましょう。

```
[COMMON]
FILES = 30
BUFFERS = 20
DOS = HIGH,UMB
SHELL = C:¥DOS¥COMMAND.COM □/P □ C:¥DOS
DEVICE = C:¥DOS¥HIMEMS.SYS
```

●ブロック

実際の環境を記述したブロックを作成しましょう。ここでは「DOSV」「PC-DOS」「WINDOWS」です。

```
[DOSV]
STACKS = 0,0
COUNTRY = 081,932,C:¥DOS¥COUNTRY.SYS
DEVICE = C:¥DOS¥EMM386.EXE RAM X = C800-CFFF
DEVICE = C:¥DOS¥$FONT.SYS □/P = C:¥FONT
DEVICEHIGH = C:¥DOS¥$DISP.SYS
```

```
DEVICEHIGH = C:¥DOS¥$IAS.SYS
DEVICE = C:¥DOS¥POWER.EXE
DEVICEHIGH = C:¥DOS¥ANSI.SYS /X
INSTALL = C:¥DOS¥IBMMKKV.EXE □/M = S □/Z =4 □/C □/
L □/J = 90 □/S = C:¥DOS¥MULTDICT.PRO □/U = C:¥$USR-
DICT.DCT

[PC-DOS]
STACKS = 0,0
DEVICE = C:¥DOS¥EMM386.EXE RAM X = C800-CFFF
[WINDOWS]
STACKS = 9,256
COUNTRY = 081,932,C:¥DOS¥COUNTRY.SYS
DEVICE = C:¥DOS¥EMM386.EXE NOEMS
DEVICE = C:¥DOS¥$FONT.SYS □/P = C:¥FONT
DEVICEHIGH = C:¥DOS¥$DISP.SYS
DEVICE = C:¥DOS¥POWER.EXE
```

● INCLUDEコマンドを活用する

ここで「DOSV」と「WINDOWS」のブロックには共通の記述が多く見受けられます。次の3行がまったく同じです。

```
DEVICE = C:¥DOS¥$FONT.SYS □/P = C:¥FONT
DEVICEHIGH = C:¥DOS¥$DISP.SYS
DEVICE = C:¥DOS¥POWER.EXE
```

ま、これくらいならとくに重複していてもかまいませんが、PCMCIAカードなどのドライバ類まで組み込んだりすると、重複する部分が相当多くなってきます。その場合は、重複する部分をひとまとめにしたブロックを作成し、INCLUDEコマンドでそのブロックを必要な箇所に読み込むことができます。

たとえば、ここでは先の3行を[JP_DISP]というようなブロックにして

おき、必要なところに「INCLUDE = JP_DISP」と記述しておけばいいのです。つまり、次のようなCONFIG.SYSになります。

```
[COMMON]
FILES = 30
BUFFERS = 20
DOS = HIGH,UMB
SHELL = C:¥DOS¥COMMAND.COM □/P □ C:¥DOS
DEVICE = C:¥DOS¥HIMEMS.SYS

[DOSV]
STACKS = 0,0
COUNTRY = 081,932,C:¥DOS¥COUNTRY.SYS
DEVICE = C:¥DOS¥EMM386.EXE RAM X = C800-CFFF
DEVICE = C:¥DOS¥$FONT.SYS □/P = C:¥FONT
INCLUDE = JP_DISP
DEVICEHIGH = C:¥DOS¥ANSI.SYS □/X
INSTALL = C:¥DOS¥IBMMKKV.EXE □/M = S □/Z = 4 □/C □/
L □/J = 90 □/S = C:¥DOS¥MULTDICT.PRO □/U = C:¥$USR-
DICT.DCT

[PC-DOS]
STACKS = 0,0
DEVICE = C:¥DOS¥EMM386.EXE □ RAM □ X = C800-CFFF

[WINDOWS]
STACKS = 9,256
COUNTRY = 081,932,C:¥DOS¥COUNTRY.SYS
DEVICE = C:¥DOS¥EMM386.EXE □ NOEMS
INCLUDE = JP_DISP

[JP_DISP]
DEVICE = C:¥DOS¥$FONT.SYS □/P = C:¥FONT
```

```
DEVICEHIGH = C:¥DOS¥$DISP.SYS
DEVICE = C:¥DOS¥POWER.EXE
```

● AUTOEXEC.BATを作る

せっかくCONFIG.SYSで環境を切り替えるのですから、AUTOEXEC.BATも各環境に対応したものでなければなりません。

まず、そのキーとなるのは「CONFIG」という環境変数です。マルチCONFIGでメニューを選択すると、そのメニューのブロック名が「CONFIG」という環境変数にセットされます。

たとえば、起動時に表示されるメニューから「Japanese Mode」を選択すると、「MENUITEM = DOSV,Japanese　Mode」というメニューの「DOSV」が環境変数「CONFIG」にセットされるのです。つまりここで言えば「SET CONFIG = DOSV」と指定したのと同じこととなっています。同様に「English Mode」を選択すれば「PC-DOS」が、「Windows3.1」を選択すれば「WINDOWS」が環境変数「CONFIG」にセットされます。

ということは、AUTOEXEC.BATで環境変数「CONFIG」を参照すれば、その内容によって処理を分岐させることができるということです。そのためには、「IF～処理」という分岐コマンドを使わなければなりません。さっそく作ってみましょう。

■ 分岐ルーチン

環境変数の内容を参照して分岐する方法は2つあります。一つは「IF」コマンドで環境変数を参照し、分岐する方法。もう一つは直接環境変数をジャンプ先のラベルとしてしまう方法です。

1．IFコマンドを使う方法

IFコマンドで環境変数を参照する場合は、環境変数の前後に「%」をつけさらに「" "」でくくります。

また、比較演算子は「==」と=を2つ並べなければなりません。一つですと代入になってしまいます。

```
@ECHO OFF
IF "%CONFIG%" == "DOSV" GOTO DOSV
IF "%CONFIG%" == "PC-DOS" GOTO PC-DOS
IF "%CONFIG%" == "WINDOWS" GOTO WINDOWS
```

1行目（@ECHO　OFFをのぞいて）を例にとると、もしも「CONFIG」の内容が「DOSV」であるなら、ラベル「DOSV」にジャンプせよというものです。

2．直接ラベル先とする

環境変数の内容をジャンプ先として参照することもできます。

```
@ECHO □ OFF
GOTO □%CONFIG%
```

つまり「CONFIG」の内容を参照し、それをラベル名としてジャンプするという処理も可能なのです。たとえば「CONFIG」に「DOSV」が入っているなら、この一行は「GOTO DOSV」と解釈されます。

どちらの方がいいかは一長一短ですが、見てもわかるように後者の方が簡潔で作成するにも楽ちんです。

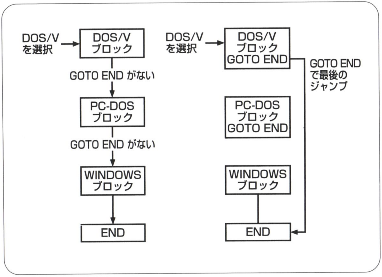
■ 各モジュールを作る

どちらの方法にしろ、分岐先のブロックを作成しておかなければ意味がありません。モジュールはバッチファイルのラベルとなりますから、ここでは「DOSV」「PC-DOS」「WINDOWS」というラベルをつけてそこに必要な起動手順を記述しておきます。ラベルにはこのように必ず頭に「：」をつけてください。また各モジュールには必ずそのモジュールの最後にバッチファイルから抜けれるように、バッチファイルの最後にジャンプするような処理を付け加えておかなければなりません。AUTOEXEC.BATは頭から[5]**シーケンシャル**に実行されるものですから、先頭からせっかく分岐してもその後は連続的に1行ずつ実行されてしまうのです。

[5]「連続して」という意味。

したがって、その処理がなされていない場合、たとえば「DOSV」を選択したとしても、次に「PC-DOS」のブロック「WINDOWS」のブロックが実行されてしまいます（図18）。

◎図18　AUTOEXEC.BATはシーケンシャルに実行される



以上のことを頭に入れた各モジュールは次のようになります。

```
:DOSV
PROMPT □ $P$G
PATH □ C:¥DOS;
SET □ TEMP = C:¥DOS
C:¥THINKPAD¥FUELDOS
C:¥DOS¥SMARTDRV.EXE
LH C:¥DOS¥KEYB.COM □ JP,932,C:¥DOS¥KEYBOARD.SYS
GOTO □ END

:PC-DOS
PROMPT □ $P$G
PATH □ C:¥DOS;
```

```
SET □ TEMP = C:¥DOS
C:¥DOS¥SMARTDRV.EXELH □ C:¥DOS¥KEYB.COM □ JP,
932,C:¥DOS¥KEYBOARD.SYS
GOTO END

:WINDOWS
PATH □ C:¥WINDOWS;C:¥DOS;
SET □ TEMP = C:¥DOS
SET □ IBMAV = C:¥DOS
C:¥THINKPAD¥FUELDOS
C:¥DOS¥SMARTDRV.EXE
LH C:¥DOS¥KEYB.COM □ JP,932,C:¥DOS¥KEYBOARD.SYS
WIN

:END
```

これで一応まともには動きますが、今度も共通の部分がだいぶあります。これくらいならかまわないという感じもしますが、もっと簡潔にわかりやすいようにするには、共通部分を分岐ルーチンの前に書くようにしなければなりません。

また、完全に共通ではないものの、PATH などは「C:¥DOS」だけであろうが、「C:¥DOS;C:¥WINDOWS;」であろうとあまりかわりがないので、「C:¥DOS;C:¥WINDOWS;」として共通のモジュールとしておいた方がいいでしょう。

```
:*******
:共通のルーチン
:*******
@ECHO □ OFF
PATH □ C:¥WINDOWS;C:¥DOS;
C:¥THINKPAD¥FUELDOS
C:¥THINKPAD¥FUELDOS
```

```
SET □ TEMP = C:¥DOS
LH □ C:¥DOS¥KEYB.COM □ JP,932,C:¥DOS¥KEYBOARD.
SYS
:******
:分岐ルーチン
:******
GOTO □ %CONFIG%
:***************
:日本語モードのためのモジュール
:***************
:DOSV
PROMPT □ $P$G
C:¥DOS¥SMARTDRV.EXE
GOTO □ END
:**************
:英語モードのためのモジュール
:**************
:PC-DOS
PROMPT □ $P$G
C:¥DOS¥SMARTDRV.EXE
GOTO □ END
:***************
:WINDOWS のためのモジュール
:***************
:WINDOWS
C:¥DOS¥SMARTDRV.EXE
WIN
:***********
:終了処理のためのラベル
:***********
:END
```

日本語・英語モードの切り替えについて

DOS/Vでは、日本語と英語モードを切り替えて利用することができます。その切り替え方は、Microsoft製のMS-DOSとPC-DOSで違います（画面65）。

◎画面65　日本語モード

```
C:¥>DIR /W

 ドライブ C のボリュームは TP-C        です
 ボリューム・シリアル番号は 1ED6-0517 です
 ディレクトリーは  C:¥

[AIR]          [AS]           [ATOK]         [ATOK9]        [BAT]
[CAPIT]        [CW]           [DOS]          [ED]           [EXCEL5]
[EZ]           [FM]           [FONT]         [GAME]         [HT]
[IBMVESA]      [ICON]         [ICONDOIT]     [ICONMAKE]     [JSLIB]
[JSTOOL]       [LHA]          [MDEV]         [MIW]          [MSDOS]
[MSOFFICE]     [MYFAX]        [NU]           [NW]           [ORGANIZE]
[PKARC]        [POWER2]       [PTF310B]      [QEMM]         [QV10]
[SFW]          [SKW]          [SYS]          [SYSBAK]       [TAROWIN6]
[TBACK]        [TEST]         [TP]           [VCAP]         [WIN32APP]
[WINDOWS]      47.TXT         AUTOEXEC.BAT   AUTOEXEC.0     AUTOEXEC.TMP
COMMAND.COM    CONFIG.0       CONFIG.SYS     CONFIG.TMP     KAN.BAT
P.BAT
         56 個              64,111 バイトのファイルがあります
                        26,898,432 バイトが使用可能です

C:¥>_

     英数 半角   R
```

●PC-DOSの場合：CHEVコマンドを使う

PC-DOSでは、DOS/Vモードでマシンを起動していれば、CHEVコマンドを使って簡単にモードを変更することができます。日本語モード・英語モードへの切り替え方法は次のようになります。

日本語モード→英語モード

```
CHEV □ US ↵
```

英語モード→日本語モード

```
CHEV □ JP ↵
```

● MS-DOS (Microsoft) の場合：CHEV、CHCP、US・JPコマンドを使う

こちらではPC-DOSと同じようにCHEVコマンドを利用する方法、CHCPコマンドを使う方法、US・JPコマンドを使う方法があります。

日本語モード→英語モード

```
CHEV □ US ↵
CHCP □ 437 ↵
US ↵
```

英語モード→日本語モード

```
CHEV □ JP ↵
CHCP □ 932 ↵
JP ↵
```

切り替えることができる条件は、BILING.SYSが組み込まれていて、かつNLSFUNC.EXEが実行されていることが必要です。

● PC-DOSでのNLSFUNC.EXE

PC-DOSにもNLSFUNC.EXEがあります。このコマンドを実行すると、Microsoft製MS-DOSと同じように**CHCPコマンド**でも英語モードに変更することができます。ただし、**CHEVコマンド**と違ってどうも中途半端なモード切替のようです。画面表示などはほとんど変わりませんし、日本語もちゃんと表示することができます。

これではほとんど役にたちませんので、CHEVコマンドで変更した方がいいでしょう。

●完全に切り替える場合：SWITCH コマンド

CHEV コマンドや CHCP では、単に国別の情報にあった表示を行ったり画面のモードを変更するだけです。つまりあくまでも基本は DOS/V モードであって、表示だけ英語モードとなっているという感じです。これでは本来の英語モードで動作するソフトウェアなどを実行すると不具合が生じることがあります。特に、ゲームソフトなどではその傾向が顕著です。

完全にモードを切り替えるには、「SWITCH」コマンドを使います。このコマンドでは何もオプションは必要ありません。単に「SWITCH ↵」で日本語モードが英語モードに、英語モードが日本語モードに変更されます(画面66)。

◎画面 66 SWITCH コマンドでの変更

```
C:¥>SWITCH
[注意]: モード切り換え中は(PS/2モードに切り換わるまで) 電源を切らないでください。
システムを、PS/2モード (英語モード) に切り換えます。
システムを切り換えますか (Y/N) ? _

英数 半角  R
```

実は、この SWITCH コマンドは本当のコマンドではなく、DOS のディレクトリにあるバッチファイル**「SWITCH.BAT」**です。**HARDBOOT コマンド**により日本語モード用の CONFIG.SYS、AUTOEXEC.BAT を英語モード用の CONFIG.SYS と AUTOEXEC.BAT に書き換え（またはその逆）マシンをリセットしているのです。

したがって、このおいしい部分だけをとりだせば、SWITCH コマンドは必要ないということになります。たとえば、現在日本語モードで動作して

いて、これを英語モードに変更するのであれば、次のように操作してください。

```
HARDBOOT □ PS52 □ PS22 ↵
```

逆の場合は「PS52　PS22」を「PS22　PS52」としてください。この「PS22」と「PS52」というのは、HARDBOOTコマンドのオプションで、それぞれ「英語モード」と「日本語モード」を表しています。「PS52 PS22」と「PS22 PS52」ではそれぞれ次のような作業が行われます。

「PS52 PS22」

```
CONFIG.SYS → CONFIG.PS5　へ名前変更
AUTOEXEC.BAT → AUTOEXEC.PS5　へ名前変更
CONFIG.PS2 → CONFIG.SYS　へ名前変更
AUTOEXEC.PS2 → AUTOEXEC.BAT　へ名前変更
マシンをリセット
```

「PS22 PS52」では

```
CONFIG.SYS → CONFIG.PS2　へ名前変更
AUTOEXEC.BAT → AUTOEXEC.PS2　へ名前変更
CONFIG.PS5 → CONFIG.SYS　へ名前変更
AUTOEXEC.PS5 → AUTOEXEC.BAT　へ名前変更
マシンをリセット
```

これを見ると、英語モードでは「PS2」という拡張子のCONFIG.SYSとAUTOEXEC.BATを使い、日本語モードでは「PS5」という拡張子のものを使うということになっています。したがって、PS2の拡張子のCONFIG.SYSとAUTOEXEC.BATを編集すれば、英語モードの環境構築を行うことができます。もちろん、編集方法やコマンドの使用方法などは、日本語DOS/V環境の構築と同じです。

```
BUFFERS = 20
FILES = 30
DOS = HIGH
COUNTRY = 001,437,C:¥DOS¥COUNTRY.SYS
SHELL = C:¥DOS¥COMMAND.COM □/P □/E:512
DEVICE = C:¥DOS¥HIMEM.SYS
DEVICE = C:¥DOS¥EMM386.EXE RAM
DEVICE = C:¥DOS¥SETVER.EXE
DEVICE = C:¥DOS¥ANSI.SYS □/X
REM DEVICEHIGH = C:¥DOS¥$FDD12.SYS
DEVICE = C:¥DOS¥POWER.EXE
```

HARDBOOT コマンドにオプションを付けず、単に「**HARDBOOT ↵**」とするとパソコンをリセットすることができますが、現在が日本語モードの時に「**HARDBOOT □ PS22 □ PS52 ↵**」とやっても何も起こりません。同様に英語モードの時に「**HARDBOOT □ PS52 □ PS22 ↵**」とするのも無効です。このようにオプションをつけた場合、HARDBOOT コマンドはそのオプションの内容を理解して、無用のリセットを行わないようになっています。

なお、SWITCH コマンドはバッチファイルですから、他のプログラムに「SWITCH.*」というようなプログラムが存在する場合、PATH に指定するディレクトリ順によっては SWITCH コマンドが利かなくなることもあります。その場合は、「**C:¥DOS¥SWITCH ↵**」とすれば SWITCH コマンドを起動することができます。

なお、MS-DOS の SWITCH コマンドはバッチファイルではなく SWITCH.EXE というプログラムです。そちらでは CONFIG.US（JP）を CONFIG.SYS に AUTOEXEC.US（JP）を AUTOEXEC.BAT に変更して環境を切りかえます。

第3部
実例！周辺機器設定

DOS/V マシンは非常に柔軟にシステムをグレードアップすることができますが、周辺機器を接続して DOS/V で利用するには、その周辺機器を制御するデバイスドライバを組み込まなければなりません。ここでは、周辺機器を接続するための基礎知識と、現在広く使用されている周辺機器を DOS/V マシンに接続する実例をご紹介します。

もちろん、デバイスドライバを組み込んで周辺機器を増設すれば、全体のメモリ構成も変化してきますので、メモリのカスタマイズも欠かせませんので、メモリカスタマイズユーティリティの利用方法も解説します。

第1章
実例！周辺機器のセットアップ

環境構築が必要なケースとして「周辺機器を増設したとき」という1項目があったのを憶えていらっしゃいますでしょうか？　周辺機器を増設すれば、ハードウェア環境が変わってしまいますので、当然ソフトウェア的にも環境を構築し直さなければなりません。また、ハードウェアによっては、その機器に適応するようなソフトウェア設定を行わなければ機器自体動作しないようになっています。

なお、基本となるCONFIG.SYSとAUTOEXEC.BATはThinkPadのものとまったく同じものを使用します。

```
<CONFIG.SYS>
BUFFERS=20
FILES=30
DOS=HIGH,UMB
COUNTRY=081,932,C:¥DOS¥COUNTRY.SYS
SHELL=C:¥DOS¥COMMAND.COM □/P □ C:¥DOS
DEVICE=C:¥DOS¥HIMEM.SYS
DEVICE=C:¥DOS¥EMM386.EXE □ RAM
DEVICEHIGH=C:¥DOS¥$FONT.SYS □/P=C:¥FONT
DEVICEHIGH=C:¥DOS¥$DISP.SYS
DEVICEHIGH=C:¥DOS¥$IAS.SYS
DEVICEHIGH=C:¥DOS¥ANSI.SYS □/X
INSTALL=C:¥DOS¥IBMMKKV.EXE□/M=S□/Z=4□/C□/L□/J
=90□/S=C:¥DOS¥MULTDICT.PRO□/U=C:¥$USRDICT.DCT
```

```
STACKS = 9,256
LASTDRIVE = Z

<AUTOEXEC.BAT>
@ECHO □ OFF
SET □ COMSPEC = C:¥DOS¥COMMAND.COM
PROMPT □ $P$G
PATH □ C:¥;C:¥WINDOWS;C:¥DOS;
SET □ TEMP = C:¥DOS
C:¥DOS¥SMARTDRV.EXE
LH□C:¥DOS¥KEYB.COM□JP,932,C:¥DOS¥KEYBOARD.SYS
```

この環境では画面67のようなメモリ状況となっています。

◎画面 67　MEM コマンドでの確認

```
C:¥>MEM

メモリーのタイプ     合計  =  使用済 +  使用可
----------------  -------   -------   -------
基本                 640K       71K      569K
UMB                   91K       77K       13K
アダプタ RAM/ROM     384K      384K        0K
XMS* メモリー     15,269K    4,428K   10,841K
----------------  -------   -------   -------
システム全メモリ  16,384K    4,960K   11,424K

1MB 以下全メモリ     731K      148K      583K

全 EMS メモリー                     15,680K (16,056,320 バイト)
使用可能 EMS* メモリー              11,040K (11,304,960 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ          569K (582,608 バイト)
使用可能最大 UMB メモリー              13K  (13,600 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

I/O ポート、IRQ、ROM アドレス、「DMA」

周辺機器をセットアップし環境を構築する場合、常に問題となるのは「I/Oポート」「IRQ」「ROM アドレス」「DMA」です。

● I/O ポート

周辺機器とパソコン本体とがデータをやりとりする際に使用する窓口のようなものです。たとえば、[1]**SoundBlaster16ASP** のデフォルトセッティングでは、220H〜233H を使用し、さらに[2]**MIDI** を利用する場合は 330H〜331H までを使用するとなっています。

● IRQ (Interrupt ReQuest)

ハードウェア割り込みを利用してアクセスする周辺機器には、IRQ を割り当てる必要があります。IRQ はさまざまな機器が使用していますが、DOS/V マシンには8チャンネルずつのコントローラが2個用意されていますので、合計16チャンネル（0〜15）使用することができますから、まだ余裕があります。デフォルトでは表6のような IRQ 状況となっています。

[1]サウンド機能を提供するサウンドカード。DOS/V ではほぼスタンダードとなっている。
[2]電子楽器のデータ転送方式について定めた規格 Musical Instrumaut Digital Interface

◎表6　IRQ一覧

IRQ	デフォルト割り当て	摘要
0	システムタイマー	
1	キーボード	
2	使用可能	カスケード（8～15）
3	COM2	COM4と共用
4	COM1	COM3と共用
5	プリンタ（LPT2）	サウンドブラスターに割り当てることが多い
6	FDDコントローラ	
7	プリンタ	
8	リアルタイムクロック	
9	使用可能	
10	使用可能	
11	使用可能	SCSIインタフェイスが利用する
12	使用可能	
13	数値演算コプロセッサ	
14	HDDコントローラ	
15	使用可能	

8bitカードでは0～7までしか使えない

SoundBlasterではIRQの5番がデフォルトとなっています。

●DMA（Direct Memory Access）チャンネル

「DMA」とはCPUを介さず、拡張カードとメモリが直接データをやりとりする方式ですが、そのためにDMAチャンネルを割り当てます。

昔パソコンのCPUが低速だった頃はCPUの負担を軽減するDMAが盛んに使われましたが、現在ではCPUの能力が高いのでかえってDMAを使用する方が速度が落ちてしまいますので、高速なデータ転送が必要な周辺機器ではDMAを使用しないようになっています。SoundBlasterでは、8-bitサウンド用にDMAの1番を、16-bitサウンド用に5番を利用しますが、デフォルトでは表7のような状況です。

◎表７　DMA チャンネルの利用状況

DMA		摘要
0	利用可能	
1	利用可能	SoundBlaster 8bit
2	ディスクコントローラ	
3	利用可能	
4	カスケード（0～3）	
5	利用可能	SoundBlaster 16bit
6	利用可能	
7	利用可能	

● ROM アドレス

拡張ボードには ROM にプログラムが焼き付けられているものもありますが、この ROM に直接アクセスするのではなく、メモリの拡張 ROM エリア（640K バイト～1M バイト）にアクセス用の窓を設けてアクセスします。

たとえば、ADAPTEC の SCSI カードでは、デフォルトとして「CD000」が ROM アドレスとして割り当てられます。

●設定

「I/O ポートアドレス」「IRQ」「ROM アドレス」「DMA」の設定は、各周辺機器によって違います。ボード上のジャンパスイッチで指定するものもありますし、PCMCIA カードのようにソフトウェアで指定するものもあります（画面68）が、とにかく「I/O ポートアドレス」「IRQ」「ROM アドレス」「DMA」が、他の機器が使用しているものと重複しないように設定しなければなりません。

◎画面68　ETHERNETカードのiniファイル

```
C:¥NW>TYPE NET.CFG
# Setup the TDKCD02 card
Link driver TDKCD02

        INT 5
        PORT 300
        MEM C8000
        SLOT 1
        Frame ETHERNET_802.3

C:¥NW>_
```

たった一つだけ周辺機器を増設するならあまり問題はないでしょうが、複数の機器を増設する場合はへたをするとわけがわからなくなりますから、面倒でも表8のように一覧表にしておくことをおすすめします。

◎表8　一覧表

	IRQ	DMA	I/O	ROM
SoundBlaster	5	1,5	220-233H	
MIDI			330-331H	
ETHERNET	10		300H	D0000
SCSI	11	0	340H	DC000

また、MSDコマンドの起動画面で Q を押すと、IRQの状態を表示することができます（画面69）。ただしこれはあくまでも目安としておいてください。

◎画面69　MSDでIRQの状況を表示させる

```
F/ﾌｧｲﾙ U/ﾕｰﾃｨﾘﾃｨ H/ﾍﾙﾌﾟ
                        IRQの状態
IRQ  アドレス   ﾃﾞｨｽｸﾘﾌﾟｼｮﾝ       検出               処理
---  ---------  ---------------  -----------------  ---------------
  0  0BE2:1875  タイマ クリック   Yes                ﾌﾞﾛｯｸ ﾃﾞﾊﾞｲｽ
  1  1093:05C2  キーボード        Yes                VGACAP.COM
  2  0484:0057  2つ目の8259A      Yes                ﾃﾞﾌｫﾙﾄ ﾊﾝﾄﾞﾗ
  3  0484:006F  COM2: COM4:       COM2:              ﾃﾞﾌｫﾙﾄ ﾊﾝﾄﾞﾗ
  4  0484:0087  COM1: COM3:       COM1:              ﾃﾞﾌｫﾙﾄ ﾊﾝﾄﾞﾗ
  5  0484:009F  LPT2:             No                 ﾃﾞﾌｫﾙﾄ ﾊﾝﾄﾞﾗ
  6  0484:00B7  ﾌﾛｯﾋﾟｰ ﾃﾞｨｽｸ      Yes                ﾃﾞﾌｫﾙﾄ ﾊﾝﾄﾞﾗ
  7  0070:06FE  LPT1:             Yes                システム領域
  8  0484:0052  ﾘｱﾙ ﾀｲﾑ ｸﾛｯｸ      Yes                ﾃﾞﾌｫﾙﾄ ﾊﾝﾄﾞﾗ
  9  F000:EEB0  リダイレクトIRQ2  Yes                BIOS
 10  0484:00CF  (予約)                               ﾃﾞﾌｫﾙﾄ ﾊﾝﾄﾞﾗ
 11  0484:00E7  (予約)                               ﾃﾞﾌｫﾙﾄ ﾊﾝﾄﾞﾗ
 12  0484:00FF  (予約)                               ﾃﾞﾌｫﾙﾄ ﾊﾝﾄﾞﾗ
 13  F000:EEB9  数値ｺﾌﾟﾛｾｯｻ       Yes                BIOS
 14  0484:0117  ハード ディスク   Yes                ﾃﾞﾌｫﾙﾄ ﾊﾝﾄﾞﾗ
 15  F000:9714  (予約)                               BIOS

                               OK

IRQの状態: ハードウェア割り込みの状態の表示
```

SCSIカードを増設する

SCSIカードを増設した場合、かならずSCSI用のデバイスドライバをインストールしなければなりません。このデバイスドライバはSCSIカードに附属していますので、SCSIカードを購入時に確認してください。もしも不幸にしてSCSIカードにデバイスドライバが附属していない場合は、汎用タイプのSCSIドライバも販売されていますからSCSIカードと一緒に購入してください。

●カード上での設定

SCSIでは、カード上のジャンパスイッチで設定を行うことが多く、さらにデバイスドライバのオプションで設定を変更することも可能です。

たとえば、ADAPTECのAHA-1520では「IRQ」「ROMアドレス」「DMA」その他を5つのジャンパスイッチ群で設定することができます。デバイスドライバで「I/Oポートアドレス」(デフォルトは340h) やDMAチャンネルを変更することができるものもありますが、できるだけジ

ャンパスイッチで変更を行うようにしてください。

もちろん、他の周辺機器とコンフリクト（衝突）を起こさないように設定を行わなければなりません。

IRQ

通常 SCSI は IRQ11 を使用します。EZ-SCSI では SCSI カードのジャンパスイッチの設定をそのままデフォルトとして使います。他の機器と重なる恐れがなければ変更する必要はないでしょう。

IO

ADAPTEC の SCSI カードでは、機種によってデフォルトの IO アドレスが違います。また、選択できるアドレスもさまざまです。やはり他の周辺機器と重ならないように設定してください。とくに SoundBlaster で MIDI 機能を使用するときは、こちらで IO アドレスを変更する必要があるかもしれません。

DMA

DMA は最近あまり使われなくなりましたが、もしも他の周辺機器でも使用している場合は、その DMA 番号と別の番号を指定しなければなりません。

ADAPTEC の SCSI カードでは、デフォルトで DMA の 0 番を使用するようにジャンパスイッチで設定してあります。選択できるのは「0、5、6、7」ですが、0 番で問題はないはずです。

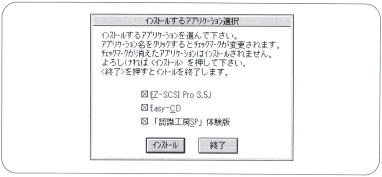
●デバイスドライバの組み込み

現在広く使用されている SCSI カードは ADAPTEC 製のものですが、ADAPTEC の SCSI カードにはドライバ類が附属しています。もちろんドライバ類をインストールするためのインストーラも附属していますから、そのインストーラを使えばドライバやユーティリティを簡単にセットアップすることができます。

最新の ADAPTEC の SCSI カードには「EZ-SCSI」というユーティリティが附属していますが、この「EZ-SCSI」はウィンドウズからインスト

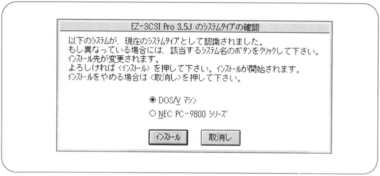
ールするようになっています（画面70）。

◎画面 70　ADAPTEC のドライバをインストールする

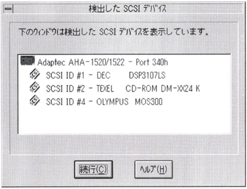


起動後はほとんど自動的にインストールが行われます。使用するマシンのシステムタイプやインストールされている SCSI 機器が自動的にスキャンされます（画面71、画面72）。さらにスキャンされた SCSI カードの設定を検出し、その設定でインストールを行います。

◎画面 71　マシンタイプ

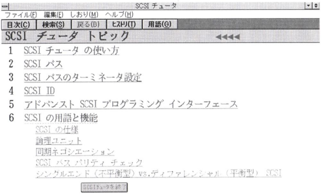


◎画面 72　SCSI 周辺機器が検出された



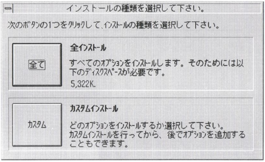
また、「チュートリアル」という名前のオンラインヘルプが充実していますので、初めて SCSI カードをセットアップするユーザーでも迷わずにインストールを行うことが可能です（画面73）。

◎画面 73　チュートリアル



もちろん全自動といっても、必要最低限の選択はしなければなりません。全自動でインストールするかカスタムインストールするかも選択できます（画面74）。

◎画面 74　インストールの方法を選択する

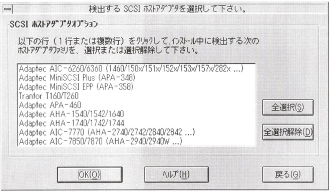


インストールが終了したら、新しい CONFIG.SYS と AUTOEXEC. BAT を見てみましょう。

●カスタムインストール

EZ-SCSI のインストールは非常に簡単ですので、通常は全自動モードで行えばいいのですが、システムによってはインストーラがセットアップされている SCSI カードを検出することができないこともありますので、そのような場合はカスタムインストールで対応することができます（画面75）。

◎画面 75　カスタムインストールで SCSI カードの種類を選択する



●メモリ関係のセットアップ

SCSIのデバイスドライバやCD-ROMドライバをインストールすると、その分フリーメモリを圧縮することになります（画面76）。

◎画面 76　メモリを確認してみる

```
C:¥>MEM

メモリーのタイプ    合計  =  使用済 +  使用可
----------------  -------  -------  -------
基本                640K     150K     490K
UMB                  91K      77K      13K
アダプタ RAM/ROM    384K     384K       0K
XMS* メモリー     15,269K   4,428K  10,841K
----------------  -------  -------  -------
システム全メモリ  16,384K   5,040K  11,344K

1MB 以下全メモリ    731K     227K     503K

全 EMS メモリー                   15,680K (16,056,320 バイト)
使用可能 EMS* メモリー            11,040K (11,304,960 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ        490K (501,456 バイト)
使用可能最大 UMB メモリー            13K  (13,600 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

デバイスドライバを組み込んだのですから当然といえば当然ですが、MEMコマンドで確認するとコンベンショナルメモリがなんと79Kバイトも消費されています。できるだけ多くのコンベンショナルメモリを確保するためには、「DEVICE」をすべて「DEVICEHIGH」に書き換えることが必要です。また、AUTOEXEC.BATでロードする常駐プログラムはSMARTDRVを除き、すべてLHコマンドでロードするようにします。もちろん、すでにUMBが満杯ならこれでコンベンショナルメモリがより広くなることはありませんが、ここではまだUMBが13K空いていますので、少しはメモリ状況が改善されるはずです。

```
…CONFIG.SYS…
BUFFERS = 20
FILES = 30
DOS = HIGH,UMB
COUNTRY = 081,932,C:¥DOS¥COUNTRY.SYS
SHELL = C:¥DOS¥COMMAND.COM □/P C:¥DOS
DEVICE = C:¥DOS¥HIMEM.SYS
DEVICE = C:¥DOS¥EMM386.EXE □ RAM
DEVICEHIGH = C:¥DOS¥$FONT.SYS □/P = C:¥FONT
DEVICEHIGH = C:¥DOS¥$DISP.SYS
DEVICEHIGH = C:¥DOS¥$IAS.SYS
DEVICEHIGH = C:¥DOS¥ANSI.SYS □/X
INSTALL=C:¥DOS¥IBMMKKV.EXE□/M=S□/Z=4□/C□/L□/J=
90 □/S = C:¥DOS¥MULTDICT.PRO □/U = C:¥$USRDICT.DCT
DEVICEHIGH = C:¥SCSI¥ASPI2DOS.SYS □/D
DEVICEHIGH = C:¥SCSI¥ASPIDISK.SYS □/D
DEVICEHIGH = C:¥SCSI¥ASPICD.SYS □/D:ASPICD0
STACKS = 9,256
LASTDRIVE = Z

…AUTOEXEC.BAT…
@ECHO □ OFF
SET □ COMSPEC = C:¥DOS¥COMMAND.COM
PROMPT □$P$G
PATH □ C:¥;C:¥WINDOWS;C:¥DOS;
SET TEMP = C:¥DOS
C:¥DOS¥SMARTDRV.EXE
LH□C:¥DOS¥KEYB.COM□JP,932,C:¥DOS¥KEYBOARD.SYS
LH □ C:¥DOS¥MSCDEX.EXE □/D:ASPICD0 □/M:12 □/L:M
```

このように変更した結果を見てみましょう（画面77）。

◎画面 77　変更結果を確認してみる

```
C:¥>MEM

メモリーのタイプ     合計  =  使用済  +  使用可
----------------  -------   -------   -------
基本                 640K      139K      501K
UMB                   91K       80K       10K
アダプタ RAM/ROM     384K      384K        0K
XMS* メモリー     15,269K    4,428K   10,841K
----------------  -------   -------   -------
システム全メモリ  16,384K    5,032K   11,352K

1MB 以下全メモリ     731K      220K      511K

全 EMS メモリー                    15,680K (16,056,320 バイト)
使用可能 EMS* メモリー             11,040K (11,304,960 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ         501K (512,672 バイト)
使用可能最大 UMB メモリー             10K  (10,608 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

ちょっと「DEVICEHIGH」と「LH」をつけただけで、変更前よりも11K も増えています。SCSI ドライバを組み込む前から比べると 58K バイトは消費されていますが、これくらいならしかたないでしょう。

なお、ここではCD-ROMのドライバをインストールしていますが、もし CD-ROM を搭載していないマシンでしたら、CD-ROM ドライバは必要ありません。その場合はさらにメモリに余裕ができることになります。この ADAPTEC の場合は SCSI デバイスを自動的に検索してくれますので、CD-ROM ドライブが接続されていなければ CD-ROM ドライバは組み込まれません。

サウンドカードを増設する

現在のマルチメディアパソコンには当然サウンドカードが組み込まれていますが、ちょっと前のマシンにはサウンドカードが搭載されていないのが普通です。

よほどビジネスユースに的を絞らないかぎり、サウンドカードは必要不可欠の周辺機器となりつつあります。いや、ビジネスユースでも動画や音声を

付加したプレゼンテーションなど、インパクトを追い求めるならやはりサウンドカードが不可欠です。

SCSIインタフェイスではADAPTEC製のものがほぼ標準的な位置を占めていますが、サウンドカードではSoundBlasterがほぼ標準となっています。ゲームなどの音声機能もSoundBlaster対応として作られているものが多く、他のメーカーのサウンドカードも、「SoundBlaster互換」ということを明示しているものがほとんどです。

● SoundBlasterの設定

SoundBlasterにはさまざまな種類がありますが、そのセットアップ方法は基本的にほとんど共通です。まずはハードウェアの設定とセットアップです。

SoundBlasterはカード上にさまざまなジャンパスイッチがついていますし、ソフトウェアでも設定できるようになっていますが、できるだけデフォルトの設定を変更しないようにした方がいいでしょう。というのも、ゲームソフトなどがSoundBlasterのデフォルト設定で気持ちよく動くようになっていることが多いからです。とくに、IRQを変更してしまうとせっかくのゲームでも音が出なかったりすることもあります。

もしもSoundBlasterをデフォルトに設定しておいて、他の周辺機器とメモリなどのコンフリクションが発生する場合は、IRQだけはそのままの設定（IRQ5）にしておいてください。あとは適当に変更してもかまいません。できるかぎり他の周辺機器の方を変更した方がいいでしょう。

ただし、周辺機器によってはメモリアドレスなどを自由に変更できないものもありますし、変更できるとしてもある一定の範囲内でのみ変更できるだけですから、変更できる方を変更するというのが原則です。

SoundBlasterでMIDIを使用する場合は、SCSIカードのIOアドレスと重なることもあるので注意が必要です。

他のSoundBlaster互換サウンドカードの場合も、IRQさえ変更しなければあとは自由に設定を行うこともできますが、とりあえずデフォルトでやってみてそれでだめなら変更するという手法の方がいいでしょう。

●ソフトウェアの設定

ほとんどのサウンドカードには必要なデバイスドライバ、SoundBlasterには、SoundBlaster用のドライバや、各種ツールをインストールインストーラが附属しています。そのバージョンやセット内容によって違いますが、「SoundBlaster 16 ASP」（英語版）では、ディスクの#1をドライブにセットしそのディスクに入っている「INSTALL.EXE」を実行します。なお、このインストーラは英語モードなので、CHEVコマンドなどで英語モードに変更してから起動してください。

なお、この起動画面に表示されているように、インストールを行う前には「README.TXTを読むこと」と「キャッシュプログラムを外す」ことが推奨されています。

「README.TXT」はメイン画面で「View README File」を選択すると読むことができますし、書き込みに対してキャッシュを設定していない場合は、キャッシュの設定を解除する必要はありませんので、そのまま先に進むこともできます。一応READMEファイルを読んだら、次の手順でインストールを行ってください。

1．インストールの実行

メインメニューで「Begin Installation」を選択するとインストールが始まります。

2．I/Oアドレスの確認

インストーラがSoundBlaster上のジャンパスイッチで設定されたI/OアドレスとMIDIポートアドレスを表示します。

ここではポートアドレスの変更はできませんから、変更したい場合はいったんインストーラを終了してコンピュータの電源を切り、SoundBlaster上のジャンパスイッチで設定を変更しなければなりません。

3．ファイルの転送

SoundBlasterは、C:¥SB16というディレクトリに必要なファイルを転送します。これも変更はできません。

4．ディスク交換

ファイルの転送が始まればあとはインストーラにおまかせです。画面のメッセージにしたがって、ディスク交換を行ってください。

5．CONFIG.SYS、AUTOEXEC.BATのドライブ指定

インストーラはCONFIG.SYSとAUTOEXEC.BATを書き換えますので、どのドライブのCONFIG.SYSとAUTOEXEC.BATを書き換えるのかを指定することができますが、DOS/Vの場合はほとんど例外なくCドライブですから、これも⏎キーを押すだけでいいでしょう。

6．変更の確認

続いてAUTOEXEC.BATとCONFIG.SYSの変更箇所が表示されますので確認したら⏎キーを押して更新してください。

7．ウィンドウズの設定

Windows3.1のプログラムマネージャにSB16の各種ツールを登録するために、Windows3.1がインストールされているディレクトリを指定してください。

8．終了

これで終了です。なお、変更されたAUTOEXEC.BAT、CONFIG.SYS、WIN.INIは拡張子を「BAK」または「BK」として保存されます。メインメニューに戻りますので、「Exit」でインストーラを終了し、これまでの設定を有効にするためにマシンをリブートしてください。

●メモリ状況を確認する

SoundBlasterをインストールしたことで、どれだけメモリ消費量が増えているか確認しておきましょう（画面78）。

◎画面 78　メモリ状況を確認する

```
C:¥>MEM

メモリーのタイプ     合計  =  使用済 +  使用可
----------------   -------   -------   -------
基本                 640K       73K      567K
UMB                   91K       77K       13K
アダプタ RAM/ROM      384K      384K        0K
XMS* メモリー      15,269K    4,428K   10,841K
----------------   -------   -------   -------
システム全メモリ   16,384K    4,963K   11,421K

1MB 以下全メモリ      731K      151K      580K

全 EMS メモリー                 15,680K (16,056,320 バイト)
使用可能 EMS* メモリー          11,040K (11,304,960 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ       566K (579,968 バイト)
使用可能最大 UMB メモリー           13K  (13,568 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

メモリ消費量はたった3Kバイトです。これくらいならなんともなさそうですが、後々のことがありますので、できるだけコンベンショナルメモリは多いにこしたことはありません。

●メモリをカスタマイズする

SoundBlasterをインストールすると、CONFIG.SYSに1行、AUTOEXEC.BATに4行追加されます。

AUTOEXEC.BATの方は、環境変数のセットと通常のプログラム（常駐型ではない）だけですので、LHコマンドをつけても関係ありません。今のところメモリを消費しているのはCONFIG.SYSの方ということになります。その行をDEVICEHIGHコマンドに編集しましょう。

・**CONFIG.SYSの追加行を編集する**

```
DEVICE = C:¥SB16¥DRV¥ASP.SYS □/P:220
        ↓
DEVICEHIGH = C:¥SB16¥DRV¥ASP.SYS □/P:220
```

これで起動した後のメモリは画面79のようになります。

◎画面79　CONFIG.SYS 修正後のメモリ状況

```
C:¥>MEM

メモリーのタイプ   合計 = 使用済 + 使用可
---------------- ------- ------- -------
基本                640K     71K    569K
UMB                  91K     80K     11K
アダプタ RAM/ROM    384K    384K      0K
XMS* メモリー    15,269K  4,428K 10,841K
---------------- ------- ------- -------
システム全メモリ 16,384K  4,963K 11,421K

1MB 以下全メモリ    731K    151K    580K

全 EMS メモリー               15,680K (16,056,320 バイト)
使用可能 EMS* メモリー        11,040K (11,304,960 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ      569K (582,560 バイト)
使用可能最大 UMB メモリー          11K  (10,976 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

コンベンショナルメモリはSoundBlasterを組み込む前と同じ量に戻りました。その分UMBが2Kバイト減っています。とにかく成功でしょう。

CD-ROMをインストールする

マルチメディア時代のパソコンにはCD-ROMドライブが欠かせません。もちろん当初から内蔵されているタイプのパソコンが主流となっていますが、どの内蔵ドライブをより性能のいいCD-ROMドライブにグレードアップする必要性もあるかもしれませんし、内蔵されていないタイプでは新たにCD-ROMドライブを購入してセットアップしなければなりません。

販売されているCD-ROMは内蔵タイプと外付けタイプとに分かれます。

どちらを選択するか、多分に好みの問題がありますが、そのマシン専用としてみるのであれば内蔵タイプもなかなか便利です。汎用タイプをお望みでしたら外付けタイプに限ります。

● SCSIかどうか

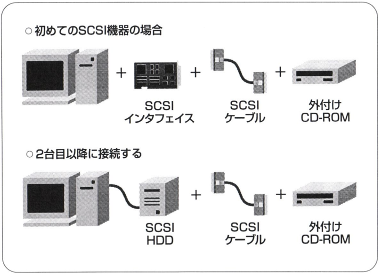
内蔵タイプにしろ外付けタイプにしろ、**インタフェイス**のタイプによって接続方法やインストール方法が違ってきますが、大きく分けてインタフェイスタイプは「SCSI」「IDE」「独自」という3つとなります。

■ SCSI

増設機器のインタフェイスとしてもっともポピュラーで使い勝手がいいのがSCSIタイプです。外付けのCD-ROMドライブはほとんどがこのタイプです。

すでにパソコンにSCSIインタフェイスカードがセットされている場合、あるいはすでに他のSCSI機器を使用している場合は、CD-ROMドライブとそれらのSCSIインタフェイスをケーブルで接続すればいいだけですので、ハードウェア的な接続は簡単です。ただし、SCSI IDの設定だけは必要です。他のSCSI機器と重複しないように設定しなければなりません(図19)。

◎図19　SCSIタイプの接続



なお、現在ではSCSI2というより高速なSCSI規格が発表されています。

これから購入するのでしたらインタフェイスはSCSI2を購入しておきましょう。

IDE

IDEインタフェイスは、DOS/Vパソコンの内蔵ハードディスク用インタフェイスとして広く使用されています。したがって、CD-ROMの接続もとりあえず簡単なのですが、IDEインタフェイスのタイプによっては複数の機器を接続することが困難な場合もありますので、このタイプを購入する場合はメーカーに確認することが必要です。

IDEでは原則的に2台のIDE機器のみ接続できるようになっていますが、拡張IDEでは4台まで接続できます。

独自

サウンドカードなどに直接接続する独自のインタフェイスを持っているものもあります。たとえば、SoundBlasterなどに接続する内蔵タイプのCD-ROMなどがこのタイプです。

独自タイプですので、汎用性に欠けるという欠点がありますが、より高機能のCD-ROMドライブが欲しい場合は、CD-ROMだけ別のタイプのものを購入してしまえばいいわけですから、とりあえずサウンドカードの購入を考えているならついでにCD-ROMもセットされたタイプの方を選択する方がいいでしょう。

● CD-ROMドライバ

どのタイプにしてもCD-ROM用のドライバが必要です。ドライバといっても、CD-ROMのドライバは2種類あって、方やハードウェア的にCD-ROMをドライブするドライバ、もう一つはソフトウェア的にCD-ROMを仮想ドライブとするドライバです。

ハードウェア的なドライバ

このドライバは、通常CD-ROMに附属しているドライバか、SCSIならSCSIカードに附属しているドライバを利用します。また、SCSIの場合は単体で販売されているSCSI関連のドライバソフトもあって、その中にも

必ずCD-ROMドライバが含まれています。

CD-ROMドライブにドライバが附属していない場合は、そのドライブは汎用のCD-ROMドライバで動作するということですから、SCSIカードのものなどをインストールすればそれで動作するでしょう。

そのインストールも、ほとんどがインストーラによってCONFIG.SYSに組み込まれますので、ユーザーが直接手を下す必要はありません。

このドライバが正常に組み込まれると、起動時に画面80のようなメッセージが表示されますので確認してください。

◎画面80　CD-ROMドライバが正常に組み込まれたメッセージ

```
Version 3.41.1
Copyright 1989-1994 Adaptec, Inc.

    HA #0 - SCSI ID 1:0 Fixed Disk - 2 logical drive(s) D: to E:

    DOS version:        6.30

1 SCSI disk drive(s) handled by ASPIDISK
2 Logical drive(s) installed
ASPIDISK.SYS Installation Successful

ASPI CD-ROM Driver for DOS
Version 3.21.3
Copyright 1990-1995 Adaptec, Inc.

   Host Adapter 0, Target SCSI ID = 2 - LUN 0: TEXEL   CD-ROM DM-XX24 K1.09

CD-ROM driver installed: 1 host adapter(s),  1 target(s)
連文節変換ルーチン(IBMMKKV.EXE Version 6.30)が EMS メモリーに導入されました。



C:¥>_
    英数 半角   R
```

ソフトウェア的なドライバ

こちらはMicrosoftから発売されているMSCDEX.EXEというプログラムです。MS-DOS（Microsoft製）には附属していますが、PC-DOSには附属していません。それでも、CD-ROMドライブ附属のソフトウェアにはほとんどの場合含まれていますので、比較的手に入れるのは容易です。

もしも、まだMSCDEX.EXEを持っていない場合は、CD-ROM購入時に販売店へ相談してみてください。なお、このMSCDEX.EXEはインタフェイスのタイプに関係なく、どのCD-ROMドライブにも必要なもの

です。

　他のドライバと一緒の場合はMSCDEXのインストールもインストーラが行ってくれますが、自分でインストールしなければならないこともあります。MSCDEXはAUTOEXEC.BATで組み込みます。形式とオプションは次の通りです。

```
LH □パス¥MSCDEX □オプション

オプション（とりあえず必要なもの）
  /D:      ：ドライバの名前を指定する
  /E       ：セクタバッファにEMSメモリを使う
  /K       ：漢字のCD-ROMボリュームを認識させる
  /L       ：CD-ROMに割り当てるドライブ名を指定する
  /M       ：セクタバッファの数
  /S       ：ネットワーク上でCD-ROMを共有する
  /V       ：起動時にメモリ状況を表示する
```

　ここで注意するのは「/D:」で指定するドライバ名です。これは、ハードウェアのドライバで指定したドライバ名と同じであることが必要です。ここの名前が違っているとCD-ROMドライブが使えなくなってしまいます。たとえば、CONFIG.SYSで組み込まれているCD-ROMドライバに次のように記述されている場合は、そのドライバ名を（プログラム名ではない）を指定します。

```
DEVICEHIGH = C:¥SCSI¥ASPICD.SYS□/D:ASPICD0
                                  ↑
                  このようにドライバ名を合わせる
                                  ↓
LH C:¥DOS¥MSCDEX.EXE □/D:ASPICD0□/M:12 □/L:M
```

　失敗した場合は失敗のメッセージが表示されます。

●その他のオプション

■ /M：セクタバッファの数

CD-ROMドライブは仮想ドライブですので、SMARTDRVのような通常のドライブ用のキャッシュプログラムは利用できません。そのかわりといってはなんですが、このオプションでバッファ用のメモリを確保します。

CD-ROMの動作がおかしい場合など、このバッファ数を増やすと解決することもあります。ただし、当然のことながらバッファ用のメモリを確保すれば、それだけメモリを消費することになります。バッファ一つにつき2048バイトのメモリが必要ですから、バッファをたくさん確保すればいいというものではありません。

なお、DOS6附属のSMARTDRVはCD-ROMもサポートしていますが、その場合はMSCDEX.EXEの後にSMARTDRVを組み込むようにAUTOEXEC.BATを編集してください。

```
@ECHO □ OFF
SET □ COMSPEC = C:¥DOS¥COMMAND.COM
PROMPT □$P$G
PATH □ C:¥;C:¥WINDOWS;C:¥DOS;
SET □ TEMP = C:¥DOS
LH□C:¥DOS¥KEYB.COM□JP,932,C:¥DOS¥KEYBOARD.SYS
LH □ C:¥DOS¥MSCDEX.EXE □/D:ASPICD0 □/M:12 □/L:M
C:¥DOS¥SMARTDRV.EXE  ←――――MSCDEXの後に移動する
```

これでSMARTDRVの状況を見ると、みごとにCD-ROMもキャッシングされていることがわかります（画面81）。

◎画面81　CD-ROMがキャッシングされた

```
C:¥>SMARTDRV
Microsoft SMARTDriveディスク キャッシュ バージョン 4.0
Copyright 1991,1992 Microsoft Corp.

キャッシュ サイズ: 2,097,152 バイト
Windows実行時のキャッシュ サイズ: 2,097,152 バイト

           ディスク キャッシュ情報
ドライブ  キャッシュの読み出し  キャッシュへの書き込み  バッファリング
----------------------------------------------------------
  A:       する          しない        不要
  B:       する          しない        不要
  C:       する          しない        不要
  D:       する          しない        不要
  E:       する          しない        不要
  M:       する          しない        不要

ヘルプを見るには"smartdrv /?"と入力してください。

C:¥>_


    英数 半角   R
```

なお、このオプションを省略するとデフォルト値の12が指定されたものとみなされますので、まったくバッファを用意せずにメモリを確保したい場合は、「/M:0」と指定しなければなりません。

/E：セクタバッファにEMSメモリを使う

SMARTDRVが使えるなら、バッファを利用する価値もあまりないとは言えますが、「/E」スイッチを利用すればバッファはEMSメモリにとられますので、コンベンショナルメモリに影響を及ぼすことはありません。

/K：漢字のCD-ROMボリュームを認識させる

日本でCD-ROMを利用する場合は指定しておいた方がいいでしょう。

/L：CD-ROMに割り当てるドライブ名を指定する

CD-ROMドライブを何ドライブに割り当てるかを指定することができます。ドライブ名を指定しないと、空いているドライブ中一番若いドライブ名が割り当てられます。

/S：ネットワーク上でCD-ROMを共有する

MS-NETWORKなどのネットワーク上でCD-ROMドライブを共有するためのオプションですから、ネットワークを組んでいないユーザーには関係ありません。

/V：起動時にメモリ状況を表示する

MSCDEXが起動する時に、メモリ状況を表示させることができます(画面82)。

◎画面82　起動時にメモリ状況を表示させる

```
ASPIDISK.SYS Installation Successful

ASPI CD-ROM Driver for DOS
Version 3.21.3
Copyright 1990-1995 Adaptec, Inc.

   Host Adapter 0, Target SCSI ID = 2 - LUN 0: TEXEL   CD-ROM DM-XX24 K1.09

CD-ROM driver installed: 1 host adapter(s),  1 target(s)
連文節変換ルーチン(IBMMKKV.EXE Version 6.30)が EMS メモリーに導入されました。

MSCDEX Version 2.23
Copyright (C) Microsoft Corp. 1986-1993. All rights reserved.
       Drive M: = Driver ASPICD0 unit 0
547024 bytes free memory
0      bytes expanded memory
12472  bytes CODE
2112   bytes static DATA
25014  bytes dynamic DATA
39856  bytes used


C:¥>_
    英数 半角   R
```

●2台以上のCD-ROMドライブを接続する

めったにないこととは思いますが、「せっかく2台あるんだから」ということで複数のCD-ROMを接続することも考えられないことではありません。ここへきて4倍速のCD-ROMドライブが主流となってきていますので、2倍速ユーザーが買い換えて2台のCD-ROMドライブを所持することもあるでしょう。

1台接続するのも2台接続するのも基本的には同じです。

CD-ROM ドライバ

複数の CD-ROM があれば、その数だけ CD-ROM ドライバを組み込まなければなりません。たとえば、両方とも SCSI タイプの CD-ROM で ADAPTEC の CD-ROM ドライバでサポートされているなら次のようになります。

```
DEVICEHIGH = C:¥SCSI¥ASPICD.SYS □/D:ASPICD0
DEVICEHIGH = C:¥SCSI¥ASPICD.SYS □/D:ASPICD1
```

MSCDEX

CD-ROM ドライバは CD-ROM の台数分記述しなければなりませんが、AUTOEXEC.BAT で組み込む MSCDEX はその必要はありません。そのかわり、台数分のドライバ名を記述しなければなりません。

```
LH□C:¥DOS¥MSCDEX.EXE□/D:ASPICD0□/D:ASPICD0□/M:12□/L:M
                    ^^^^^^^^^^^^^^^^^^^^^^^^
                    一台目        二台目
```

ビデオカードを変更する

ビデオカードを変更した場合、DOS/V の日本語モードではとくにデバイスドライバを組み込んだりする必要はありませんが、Windows3.1 で使用するツール類などをインストールする必要があります。

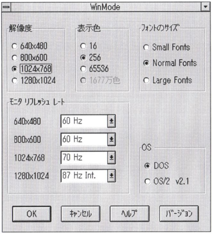
たとえ、DOS 用のツール類をインストールしたとしても、ほとんどが英語モードで動作するものですから、日本語モードで DOS の画面を高解像度にすることはできません。

PC-DOS 附属の DOSV エクステンションや、市販のスーパードライバーズを購入すれば、日本語モードでも高解像度（V-Text）を実現することができます。

●ウィンドウズユーティリティを利用する

Windows3.1用のディスプレイドライバをインストールした場合、Windows3.1上から利用できるツール類がインストールされ、画面モードをWindows3.1上から変更することが可能となるのが普通です（画面83）。

◎画面83　Windows3.1で画面モードを変更することができる



PCMCIAカードを使う

PCMCIAカードは、ノートパソコン・サブノートパソコンの拡張機器・インタフェイスとして、ここのところ急速に普及してきました。その原因として、どこでもパソコンを利用する**「モーバイルコンピューティング」**が非常に盛んになってきたということがあげられますが、「モーバイルコンピューティング」などと気取らなくても、日本の住宅事情やオフィス事情を鑑みると、ノートパソコンやサブノートパソコンが最適であることは明白ですね。

ただし、ノートタイプのパソコンにはほとんど拡張性が期待できず、それが使用上のネックとなっていました。ところが、PCMCIAという規格（もともとは増設用メモリの規格）を拡張機器のインタフェイスとして利用する

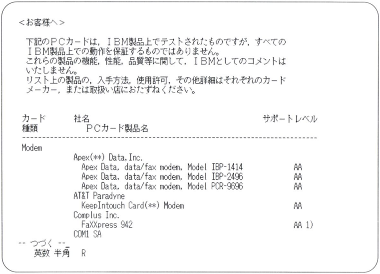
ということができるようになってきたため、最大のネックも解消されました。

現在のノートタイプパソコンは、機能的にもデスクトップにひけをとりませんし、拡張性もあるていど確保され、ファーストマシンとして利用するのに何も障害はありません。

●使えるカード、使えないカード

使用するマシンによっては、サポートしていない PCMCIA カードがあります。たとえば、ThinkPad230Cs では、**PCMCIA.CRD** というファイルでサポートされているカードの一覧を見ることができます（画面84）。

◎画面 84　サポートされているカード一覧



```
<お客様へ>

 下記のＰＣカードは，ＩＢＭ製品上でテストされたものですが，すべての
 ＩＢＭ製品上での動作を保証するものではありません。
 これらの製品の機能，性能，品質等に関して，ＩＢＭとしてのコメントは
 いたしません。
 リスト上の製品の，入手方法，使用許可，その他詳細はそれぞれのカード
 メーカー，または取扱い店におたずねください。

カード        社名                                          サポートレベル
種類             ＰＣカード製品名
-----------------------------------------------------------------------
Modem
              Apex(**) Data,Inc.
                Apex Data, data/fax modem, Model IBP-1414         AA
                Apex Data, data/fax modem, Model IBP-2496         AA
                Apex Data, data/fax modem, Model PCR-9696         AA
              AT&T Paradyne
                KeepIntouch Card(**) Modem                        AA
              Complus Inc.
                FaXXpress 942                                     AA 1)
              COM1 SA
-- つづく --
   英数 半角   R
```

マシンによってサポートしていないカードというのは、「動作しない」カードという意味ではなく「動作を保証していない」という意味です。つまり、平たく言うと、まだ試したことがないから動作するかどうかはわからない、ということですから、ここに表示されていない PCMCIA カードだからといってすぐに「使えない」と判断するべきものではありません。

PCMCIA カードは一応「規格」ですから、PCMCIA の規格に準じていて適当なデバイスドライバさえ提供されていればあとはなんとか動くはずです。

もっとも、PC-98シリーズ用とDOS/V用ではほとんど互換性がないものもありますので、PCMCIAだからといって完全に信用してしまうのは危険です。すくなくともDOS/V用が必要です。

もっと詳しくマシンとPCMCIAカードの関係について知りたい場合は、PCMCIAカードのメーカーに直接問い合わせをすれば、もう少し詳しい情報が得られるはずです。

なお、このPCMCIA.CRDで示されている「AA」とか「A」といった表示は、そのカードを使えるようにするのに、どのようなデバイスドライバを使用すればいいのかということを表しています（画面85）。

◎画面85　サポートのレベル

```
サポートレベル

A     EasyPlaying オートコンフィギュレータ(AUTODRV.SYS)でイネーブルできる
AA    EasyPlaying オートコンフィギュレータ(AUTODRV.SYS)と,OS/2 用オート
      コンフィギュレータ(AUTODRV2.SYS)でイネーブルできる
A1    EasyPlaying SRAM ドライバー(SRAMDRV2.SYS)で, リード／ライトできる
A2    EasyPlaying SRAM ドライバー(SRAMDRV2.SYS)で, リードできる
A3    EasyPlaying オートコンフィギュレータ(AUTODRV.SYS)でイネーブルできるが
      AUTODRV.INI のデフォルト設定は,
      Option=Ignore
      になっている。オートコンフィギュレータでイネーブルするには
      この行をコメントアウトする。
      ;Option=Ignore
      詳しくは, AUTODRV.INI を参照してください。
B     EasyPlaying 以外のドライバーと組み合わせて動作する。


1)    ThinkPad 720 では動作しません。

      ポイント・イネーブラ (ソケット・サービスを使わずに直接ハードウエアに
      アクセスするデバイスドライバ) をソケット・サービスと共に使用する場合は
      CONFIG.SYSのソケット・サービスの導入時に以下のオプションが必要です。
      これにより, ソケット1をポイント・イネーブラ用に予約します。
-- つづく --
   英数 半角   R
```

このPCMCIAカード一覧に名前が載っていて、サポートレベルが、A、AA、A1、A2、A3であれば、とくに他の**イネーブラ**は必要ないということになります。

● PCMCIAカードのセット

PCMCIAカードにはタイプがI～IIIまであります。IとIIは外見上ほとんど見分けはつきませんが、IIの方が若干厚みがあります。また、用途としては、Iはメモリカード専用となっていますが、IIは完全に増設用インタフ

ェイスのカードとして使用できるようになっています。

IIのスロットにセットできるカードはタイプIIに対応していなければなりません。

タイプIIIはタイプIIの2段分の厚みがあります。したがってタイプIII専用のカード（ハードディスクカード）などは、やはりカード自体もタイプIIの倍の厚みがあります。

タイプIII専用のスロットというのは少なく、タイプIIを2枚セットするかタイプIIIを一枚セットするか、両用に使えるタイプが多いようです。ThinkPadでもタイプIIが2つ上下に設けられていますので、タイプIIIのカードをセットすることも可能です。現在のノートタイプパソコンではほとんどがタイプIIIのPCMCIAスロットを装備しています。

原則的には、このソケットへカードを挿入し、そのカードと周辺機器を接続するだけでハードウェア的なセッティングは終了ですが、たまにカード上のスイッチで設定を行うものもありますので、マニュアルにしたがってください。

挿入する方向ですが、通常はカードの表が上です。ただし、カードの表を裏にしてセットするパソコンもありますので、パソコンのマニュアルで確認してください。

前後を逆にセットしようした場合は、よどの力を加えない限りソケットが壊れることはないのですが、裏表を逆にしてちょっと力を加えるとソケット内部が壊れてしまう可能性もあります。

どちらにしろあまり力を加えすぎないようにしてください。また、カードはパソコンの本体からは飛び出さないようになっていますので、パソコンのケースと面が合うまで静かに押し込むようにしてください。もちろんこれも例外があって、カードがちょっと飛び出したままという機種も存在します。

また、**X-JACK**と呼ばれるモジュラージャックを採用したモデムカードなどでは、接続するプラグが干渉して2枚のタイプII PCMCIAカードを同時に利用することができないものもありますので注意してください。

● PCMCIAカードを使うためのソフトウェア

PCMCIAカードを利用するには、原則的にソケットサービスとカードサービスというソフトウェアを組み込むことが必要です。さらにカードのイネ

ーブラとかデバイスドライバなどなど、非常に手続きが煩雑になります。

ソケットサービス

ソケットサービスと呼ばれるものです。PCMCIA ソケットを初期化し、ソケットとカードにアクセスすることを可能にします。カードによっては特別なソケットサービスを使用しなければならないものもあります。

カードサービス

ソケットとカード関連の資源を管理するドライバです。カードのセットや抜き取り、割り込みレベルなどもこのドライバで管理しています。

イネーブラ

ソケットにセットされたカードを初期化します。

こららのソケットサービスやカードサービスは、DOS に附属しているものもありますし、マシン本体に附属しているものもあります。もし、マシン本体に附属しているものがあるなら、そちらを使用する方がベターです。

現在のノートタイプパソコンには PCMCIA スロットが必ず付属していますし、DOS/V もプリインストールされていますから、ソケットサービスやカードサービスもインストール済みのものがほとんどです。

たとえば、IBM の ThinkPad230Cs では、ソケットサービスやカードサービス、さらにその他の PCMCIA 用プログラムがインストールされています。もちろん「ユーティリティディスケット」にも納められていますから、それらのファイルが壊れてしまった場合でも再インストールできるようになっています。

また、このようなソケットサービスやカードサービスを利用しないで、カード独自のイネーブラと呼ばれるソフトを利用するものもあります。したがって、PCMCIA だからといってすべてが同じように利用できるというものではなく、それぞれ利用方法が違っています。購入時に確認してください。

●PC-DOS

PC-DOSにはカードサービスやソケットサービスが附属していますが、IBM製のノートパソコン（PCMCIAスロットを搭載している機種）には、それ以外に専用のPCMCIAカード用のユーティリティが附属しています。もちろん、DOSがプレインストールされていれば、それらのユーティリティもセットアップされています。ThinkPad230Csに附属しているオリジナルのCONFIG.SYSを見るとわかりやすいでしょう。そのCONFIG.SYSの中の以下の6行が専用のPCMCIA関連のドライバです。

```
DEVICE = C:¥EZGOING¥IBMDSS01.SYS ␣/M
DEVICE = C:¥EZGOING¥IBMDOSCS.SYS
DEVICE = C:¥EZGOING¥DICRMU01.SYS ␣/MA = C800-CFFF
DEVICE = C:¥EZGOING¥$ICPMDOS.SYS
DEVICE = C:¥EZGOING¥AUTODRV.SYS ␣ C:¥EZGOING¥AUTODRV.INI
DEVICE = C:¥EZGOING¥DISKDRV.SYS
```

■ IBMDSS01.SYS＝ソケットサービス

PC-DOSに附属しているソケットサービスですが、ソケットサービスはハードウェアに依存します。

■ IBMDOSCS.SYS＝カードサービス

PC-DOSのカードサービスでは次のような機能を提供しています。

1. カード・サービス・クライアント・プログラムの登録する

2. カードの挿入と抜き取り、バッテリ状態の変化などのステータスを通知する

3. ソケット、メモリーウィンドウ、I/Oウィンドウ、割り込みレベルなどのリソースを集中管理する

DICRMU01.SYS＝リソースマップユーティリティ

マシン起動時に、カード以外の周辺機器が使用しているリソースを見つけだし、カードサービスが必要とするリソースと重複しないようにします。オプションによってカードサービスが使用するメモリ領域を指定することができます。

リソースマップユーティリティはカードサービスのクライアントとして動作しますので、カードサービスが組み込まれていないと無効です。

なお、リソースマップユーティリティには数種のオプションがあります。

/MA

PCカードに割り当てるメモリを指定するオプションです。EMSのページフレームと重複しないようにしなければなりません。たとえば、C800～CFFFまでをPCカードに割り当てるなら次のように指定します。ページフレームがD000～DFFFまでの64Kバイト使用しているなら、その範囲を避けなければなりません。たとえば、PCカード用にC800から32Kバイト割り当てるなら、次のように指定します。

```
DEVICE = C:¥EZGOING¥DICRMU01.SYS □/MA = C800-CFFF
```

このようにメモリ領域を指定する

このように指定した場合、CONFIG.SYSのEMM386.EXEのオプションでこの領域をEMSメモリに使用しないようにしなければなりません。その1行を次のように編集してください。

```
DEVICE = C:¥DOS¥EMM386.EXE □ RAM □ X = C800-CFFF
```

さらに、Windows3.1を使用している場合は、ウィンドウズのSYSTEM.INIファイルに次の1行を加えなければなりません。

```
EMMExclude = C800-CFFF
```

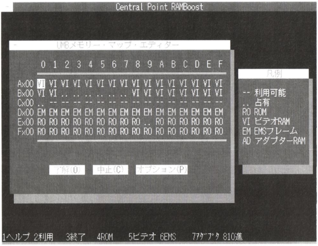
リソースマップユーティリティでは、メモリが重複しないように自動的に割り当てられますが、もしもカードの初期化がうまく行かない場合は、このメモリ指定を行うと動く場合も多いようです。

ほとんどのマシンではEMSのページフレームがD000からとられますので、他に何も周辺機器を利用していないのなら、C000～CFFFの間で適当な範囲を指定すればいいはずです。

周辺機器を利用している場合は、その周辺機器との兼合がありますが、試行錯誤でやっていくよりないでしょう。他の周辺機器のマニュアルを調べて、重複しないようにPCカードのメモリ割り当てを行ってください。

あるいは、**RAMSETUP**なども使えます。手動設定を行う「拡張(A)」を指定して表示されるメモリマップを頼りに設定することも可能でしょう。もちろんMEMコマンドでもOKです（画面86）。

◎画面86　RAMSETUPを使う



なお、PCカードによって必要なメモリ量は違っています。たとえば、モデムカードなどでしたらメモリ割り当ては不要です（0Kバイト）。もっともカードサービスが4Kバイト占有しますので、まったく必要ないというわけではありません。カードによる違いを表9にまとめておきましたので参考にしてください。

◎表９　カードによる必要メモリ量の違い

カードの種類	必要メモリ量
モデム	0Kバイト
IBM ETHERNET	16Kバイト
IBM トークンリング	24Kバイト
IBM 3270	8Kバイト
SRAM	4Kバイト
ATA（HDD）	0Kバイト

$ICPMDOS.SYS=PCMCIA パワーマネージメント・サポート・ドライバ

カードがセットされているかどうか、あるいはカードが使用されているかどうかを常に監視しているドライバです。また、擬似的なカード・サービス・イベントを送出することによって、中断による省電力機能をサポートします。

AUTODRV.SYS=オート・コンフィギュレーター

「AUTODRV.INI」に記述されている内容にしたがって、I/Oポートアドレスや割り込みレベルなどを自動的に割り当てます（画面87）。

◎画面87　AUTODRV.INIの内容

```
[Auto Configurator Option]
Beep=ON

[Modem]
CardID=MODEM
Option=UpdateBDA1

; Port1      : COM1 (3F8), COM2 (2F8), COM3 (3E8), COM4 (2E8)
; IRQ        : Interrupt Level

;* Port1=3F8,2F8,3E8,2E8,3220,3228,4220,4228,5220,5228
;* IRQ=4,3

Port1=3F8,IRQ=4
Port1=2F8,IRQ=3
Port1=3E8,IRQ=4
Port1=2E8,IRQ=3
```

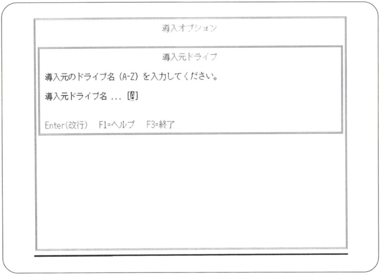
また汎用的なイネーブラとしての機能もあります。ただし、専用のイネーブラが必要なPCMCIAカードを利用する場合は、逆にじゃまになることもありますので、カードのマニュアルにしたがって**AUTODRV.INI**の内容を書き換えなければならないこともあります。

DISKDRV.SYS

メモリカードをドライブとして使う場合や、ハードディスクカードのためにドライブを割り当てるドライバです。したがって、そのようなドライブ関係のカードを使用しなければこのデバイスドライバは必要ありません。

以上のサポートソフトウェアのインストールは簡単です。「ユーティリティディスケット」をドライブにセットし、その中の**「UINSTALL.EXE」**を実行します（画面88）。

◎画面88　UNINSTALL.EXE の起動画面



●ソケットサービス、カードサービスとイネーブラ

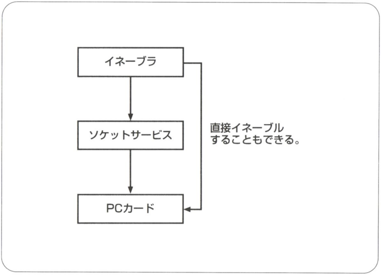
PCMCIA カードには独自のイネーブラが附属してることもあります。そのイネーブラにも2種類あり、ソケットサービスやカードサービスのクライアントとして動作するものと（オートコンフィギュレータの代わり）、ソケットサービスやカードサービスが不要なものです。後者は「ポイントイネーブラ」とも呼ばれています。また、その両方の機能のあるイネーブラもあります。

たとえば、TDK の[4]**ETHERNET** カード用のイネーブラは、図20のようにソケットサービス、あるいは直接カードに対してイネーブラすることができます。

---

[4] LAN で使用されるよい terface 規格の一種。

◎図20　TDKのイネーブラ



原則的に複数のPCMCIAカードを利用する場合は、DOS/V附属の**オートコンフィギュレータ**を利用して一気に初期化してしまった方がいいでしょう。1枚のカードのみを利用するのでしたら、ポイントイネーブラを使用する方がソケットサービスやカードサービスのドライバが不要な分、メモリの節約になります。

ThinkPad230Cs用のPCMCIAカード関係ドライバをCONFIG.SYSでインストールした場合と、それらのドライバをインストールしない場合では画面89のようにメモリの容量に大きな差ができてきます。

◎画面 89　PCMCIA ドライバはメモリを消費する

```
●PCMCIA組み込み前
メモリーのタイプ        合計  =  使用済 +  使用可
----------------     -------   -------   -------
基本                    640K      101K      539K
UMB                      63K       53K       10K
アダプタ RAM/ROM          0K        0K        0K
XMS* メモリー         10,164K    2,805K    7,359K
----------------     -------   -------   -------
システム全メモリ      10,867K    2,959K    7,908K

1MB 以下全メモリ        703K      154K      549K

全 EMS メモリー                      10,544K (10,797,056 バイト)
使用可能 EMS* メモリー                7,552K  (7,733,248 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ            539K (551,616 バイト)
使用可能最大 UMB メモリー                 6K   (5,968 バイト)
DOS は HMA 領域に常駐しています


●PCMCIA組み込み後
メモリーのタイプ        合計  =  使用済 +  使用可
----------------     -------   -------   -------
基本                    640K      169K      471K
UMB                      63K       53K       10K
アダプタ RAM/ROM          0K        0K        0K
XMS* メモリー         10,164K    2,805K    7,359K
----------------     -------   -------   -------
システム全メモリ      10,867K    3,027K    7,840K

1MB 以下全メモリ        703K      222K      481K

全 EMS メモリー                      10,544K (10,797,056 バイト)
使用可能 EMS* メモリー                7,552K  (7,733,248 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ            471K (482,096 バイト)
使用可能最大 UMB メモリー                 6K   (5,968 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

もちろん、複数のカードを使用する場合も、それぞれのイネーブラを実行すればメモリを節約することもできますが、パソコンによってはポイントイネーブラが無効な PCMCIA のソケットもありますし、イネーブル時に使用するメモリ領域などがコンフリクトする可能性も否定できませんので、複

数のPCMCIAカードを使用する場合は、安全を期してオートコンフィギュレータを使用する方がいいでしょう。たとえば、ThinkPad230Csでは下のソケット（2番）のみポイントイネーブラが有効です。

なお、ThinkPad230Csには、PCMCIAON.EXEとPCMCIAOF.EXEというプログラムが附属しています。カードサービスやソケットサービスが組み込まれた状態でPCMCIAOF.EXEを実行すると、CONFIG.SYSに記述されているそれらのドライバがすべてremアウトされ無効となります（画面90）。また、remアウトされた状態でPCMCIAON.EXEを実行すると、remが取り去られまたまたそれらのドライバが有効になります。もちろん、これらのコマンドを使用したらパソコンをリセットしなければその設定は有効になりません。

◎画面90　PCMCIA関係のドライバがremアウトされる

```
DEVICE=C:¥DOS¥EMM386.EXE RAM
DEVICEHIGH=C:¥DOS¥SETVER.EXE
DEVICEHIGH=C:¥DOS¥$DISP.SYS /HS=OFF
DEVICEHIGH=C:¥DOS¥$IAS.SYS
rem **************
rem PCMCIA DRIVER
rem **************
rem DEVICE=C:¥EZGOING¥IBMDSS01.SYS /M
rem DEVICE=C:¥EZGOING¥IBMDOSCS.SYS
rem DEVICE=C:¥EZGOING¥DICRMU01.SYS
rem DEVICE=C:¥EZGOING¥$ICPMDOS.SYS
DEVICE=C:¥DOS¥POWER.EXE
rem DEVICE=C:¥EZGOING¥AUTODRV.SYS C:¥EZGOING¥AUTODRV.INI
rem DEVICE=C:¥EZGOING¥DISKDRV.SYS
DEVICEHIGH=C:¥DOS¥$PRN.SYS
DEVICEHIGH=C:¥DOS¥ANSI.SYS /X
DEVICEHIGH=C:¥DOS¥$FDD12.SYS
REM DEVICEHIGH=C:¥DOS¥RAMDRIVE.SYS
INSTALL=C:¥DOS¥IBMMKKV.EXE /M=S /Z=4 /C /L /J=90 /S=C:¥DOS¥MULTDICT.PRO /U=C:¥$U
SRDICT.DCT
STACKS=9,256
LASTDRIVE=Z

C:¥>_
    英数 半角   R
```

●FAXモデムカードを使う

現在一番広く使用されているPCMCIAのカードというと、やはりFAXモデムカードではないでしょうか。

モーバイルコンピューティングとしてノート・サブノートパソコンを利用する場合も、また、デスクトップ機のかわりとしてデスク上で使用する場合

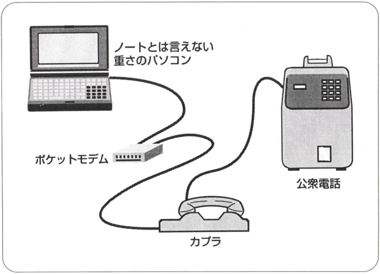
も、データ通信を行うユーザーにとってFAXモデムは非常に便利な周辺機器です。

こんな小さなカードで、14,400bpsのデータ通信／FAX通信ができてしまうことに驚きます。

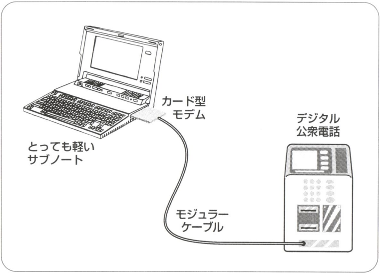
ちょっと前までは、モーバイルコンピューティングと気取ってみても、今に比べるとはるかに重いノートパソコンとポケットモデムを持って歩かなければなりませんでした。しかも、モジュラージャックがついている公衆電話などない頃はさらにカプラが必要でしたので、公衆電話からアクセスするなどというと、それこそ大仕事でした（図21）。

それに比べ、現在はもうこのFAXモデムカードをパソコンに仕込み、公衆電話がなければ携帯電話でアクセスという環境ですから、それはもう天国みたいなものですね（図22）。

◎図21　昔のモーバイルコンピューティングは大仕事



◎図22　現在はたったこれだけ



ここでは、TDKのFAXモデム「DF1414B」をThinkPad230Csで使用する手順をご紹介します。

1．どうやってイネーブルするか決める

先ほども言いましたが、PCMCIAカードをイネーブルするには、2通りの方法があります。ここではとりあえずFAXモデムだけですから、ダイレクトにイネーブルする方がいいでしょう。ただし、近々他のカードを使う予定があるなら、いっそのことここでThinkPad230Cs附属のソケットサービスとカードサービスを組み込んでしまってもいいでしょう。

ダイレクトにイネーブルする場合は、PCMCIAOF.EXEを実行して次の6行をremアウトします。また、これらのドライバを使用する場合は、remアウトされていないか確認し、remアウトされている場合はPCMCIAON.EXEを実行してそれらドライバを有効にしておきましょう。

```
DEVICE = C:¥EZGOING¥IBMDSS01.SYS □/M
DEVICE = C:¥EZGOING¥IBMDOSCS.SYS
DEVICE = C:¥EZGOING¥DICRMU01.SYS □/MA = C800-CFFF
DEVICE = C:¥EZGOING¥$ICPMDOS.SYS
DEVICE = C:¥EZGOING¥AUTODRV.SYS □ C:
¥EZGOING¥AUTODRV.INI
DEVICE = C:¥EZGOING¥DISKDRV.SYS
```

2．カードをセットする

FAX モデムカードをセットしますが、ダイレクトにイネーブルする場合は必ずソケット2（下のソケット）にセットしてください。上のソケットではダイレクトイネーブラが使えません。

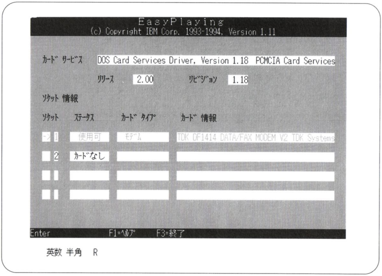
ソケットサービス、カードサービスを使用する場合は、どちらのソケットでもかまいません。カードをセットしたらパソコンをリセットしてください。

3．イネーブル

ソケットサービス、カードサービスを CONFIG.SYS でイネーブルする方法なら、AUTTODRV.SYS が起動時にモデムをイネーブルしてくれますので、起動後にイネーブラを使用する必要はありません。もうこの状態でモデムを利用することができます。

もしも不安でしたら、**EZPLAY.EXE** というプログラムでそのカードが有効かどうか見ることができます（画面91）。

◎画面91　EZPLAYで確認できる



ダイレクトにイネーブルする場合は、PCMCIAカードに附属のイネーブラを使用しなければなりません。TDKのFAXモデムでは**「TDKMODEM.EXE」**というプログラムを使います。附属のディスク(STARFAXのディスク)からこのプログラムをハードディスクにコピーしておきましょう。PATHの通っているディレクトリにコピーすると便利です。

あとは、次のようにこのイネーブラを実行すれば、TDKのモデムをイネーブルすることができます(画面92)。

```
TDKMODEM □/P □ COMx ↵
```

COMxのxには使用するポート番号を指定します。たとえばポート番号2を使用するのでしたら「**TDKMODEM □/P □ COM2 ↵**」とするだけです。もちろん、こんなことを毎回やるのは面倒ということでしたらAUTOEXEC.BATにそっくり記述してしまえばいいのです。

もしもカードがみあたらない場合は画面93のようなメッセージが表示されます。FAXモデムカードが2番のスロットに入っているかどうか、またちゃんと奥までしっかりと挿入されているかどうかを確認してください。

◎画面92　TDKのモデムがイネーブルされた

```
C:¥>AS¥TDKMODEM /P COM2
9041-0001 V3.28(C) 1993 TDK Inc. (940117 1200)

Slot 1: Modem ready on COM2.

C:¥>_

英数 半角　R
```

◎画面93　FAXモデムカードが見あたらない場合

```
C:¥>AS¥TDKMODEM /P COM2
9041-0001 V3.28(C) 1993 TDK Inc. (940117 1200)
No card(s) found.

C:¥>_

英数 半角　R
```

なお、イネーブルが成功した場合「Slot 1: Modem ready COM2」というようになっていますが、このSlot番号は気にしないでください。実際

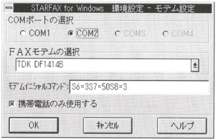
は2番のスロットでも必ず1番というように表示されるだけです。

4．ソフトウェアの設定

これでFAXモデムが使用できるようになりました。あとは通信ソフトを起動するなり、FAXソフトを使うなりしてFAXモデムを活用しましょう。ただし、通信ソフトやFAXモデムソフトで、使用するポート番号をここで指定した番号に一致させなければなりません。

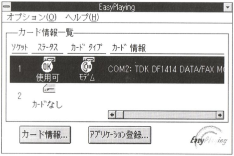
たとえば、ここで2番のポートを指定したなら、附属のSTARFAXでも2番のポートに設定しなければなりません（画面94）。

◎画面94　STARFAXでもポートを指定する



なお、Windows3.1上でもEZPLAYのユーティリティを使用することができます。そのユーティリティでFAXモデムカードが有効に使えるかどうかを確認することも可能です（画面95）。

◎画面95　STARFAXでもポートを指定する



第 *2* 章
メモリをカスタマイズする

環境構築の最終兵器はメモリのカスタマイズです。搭載されているメモリを効率よく使用し、より快適な環境を実現するためにはメモリのカスタマイズが欠かせません。

幸い DOS/V には、RAMBOOST（PC-DOS）または MEMMAKER（MS-DOS）というメモリをカスタマイズするツールが附属しています。このツールを使えば、デフォルトのメモリ設定よりも効率のよいメモリ利用が可能です。

なお、これから先の作業では、CONFIG.SYS と AUTOEXEC.BAT が書き換えられてしまいますので、必ず現在の CONFIG.SYS と AUTOEXEC.BAT をどこかのディレクトリにコピーしておいてください。

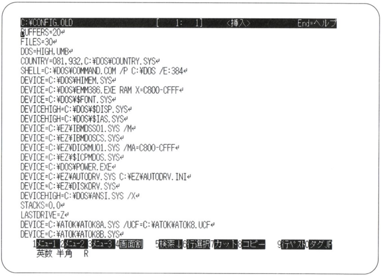
メモリカスタマイズの基本

メモリをカスタマイズするには、まずその基本テクニックを知らなければなりません。

●エディタを使う

メモリをカスタマイズするには、CONFIG.SYS や AUTOEXEC.BAT を編集しなければなりません。自動的にその修正を行ってくれるツールでカスタマイズするならいいのですが、手動で設定する場合はワープロかエディタを使用しなければなりません。もちろん手軽なのはエディタです（画面96）。

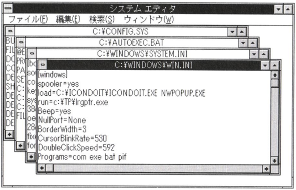
◎画面 96　エディタ（MIFES）



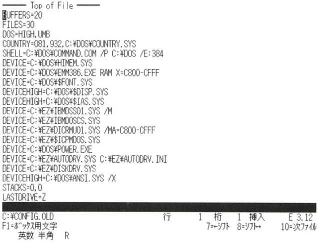
Windows3.1 のメモ帳や SYSEDIT（画面97）もいいのですが、CONFIG.SYS や AUTOEXEC.BAT を編集するために Windows3.1 を起動するというのもわずらわしい話です。

でも、DOS/V には「E」というエディタが附属していますので、使い慣れたエディタがないのであればこれを使ってください。「E」は DOS のディレクトリにありますので、**「E␣C:¥CONFIG.SYS⏎」**あるいは**「E␣C:¥AUTOEXEC.BAT⏎」**でそれらのファイルを編集することができます（画面98）。

◎画面 97 SYSEDIT



◎画面 98 E で編集を行う



●手順をつくす

CONFIG.SYSのコマンドだけで十分なメモリが確保できれば、それに越したことはありません。DEVICEHIGHコマンドを活用して、とにかくDOS/Vオリジナルの機能だけでメモリを確保するというのが基本中の基本です。

1．不要なドライバを削除する

まずやらなければならないのは、不要なドライバや常駐プログラムなどを削除することです。たとえば、ウィンドウズを中心に利用するのであれば、DOSのかな漢字変換機能は必要ないでしょうし、$IAS.SYSやANSI.SYSといったドライバも不要です。

また、プリンタドライバも不要なことが多く、さらにマウスドライバもいらないでしょう。

2．DEVICEHIGH、LOADHIGHを活用する

とにかくHIMEM.SYSとEMM386.EXEを除いて、すべてのDEVICEをDEVICEHIGHに変えてしまいましょう。

同様に、AUTOEXEC.BATの方では常駐プログラムはLHコマンドでロードするようにしてください。

3．DOSをHMAにロード

CONFIG.SYSの「DOS」コマンドを利用すると、DOS/VのシステムをHMA領域にロードすることができます。次のような一行がCONFIG.SYSに記述されているかどうか確認してください。

```
DOS = HIGH,UMB ↵ (DOS = UMB,HIGH でも可)
```

この機能を有効にするには、同時にEMM386.EXEのオプションに「RAM」が指定されているかどうかを確認しましょう。

```
DEVICE = C:¥DOS¥EMM386.EXE □ RAM ↵
```

というようになっていれば大丈夫です。

4．$FONT.SYSの位置を確認する

CONFIG.SYSに記述する順番を変更するだけで、メモリ容量が増える場合もあります。一番基本として考えなければならないのは、**$FONT.**

SYS の記述位置です。

最初の$FONT.SYS は、**INT15h** という独特のアクセス方法を採用していましたので、HYMEM.SYS よりも前に記述していました。その名残かどうか、DOS6 をインストールしても$FONT.SYS は HYMEM.SYS よりも前に記述されてしまいます。

DOS6の$FONT.SYS は他のドライバとまったく同じように UMB にロードすることもできますので、メモリのカスタマイズを行う前に、この$FONT.SYS が HYMEM.SYS の後ろに記述してあるかどうか確認しましょう。

5．STACKS コマンドを考える

STACKS コマンドでスタックを設定するとメモリが消費されます。何も指定しないとデフォルトの「**8,128**」となってしまいますから、「**STACKS = 0,0**」というように指定しておきましょう。

このように STACKS を全然用意しないと、Windows3.1 で不具合がでることもあります。その場合は「**9,256**」というように指定しておきましょう。

6．余計なドライブ

CD-ROM を搭載している機種では、CD-ROM のドライブが Q ドライブに設定されていることも多いようです。メモリ環境から見ると、不要なドライブ名はメモリを浪費するだけのものですから、もっと前のドライブ名を割り当てた方がいいでしょう。たとえば、内蔵ハードディスクが 1 ドライブ、外付けのハードディスクが 1 ドライブそれに CD-ROM というような構成でしたら CD-ROM は E:ドライブでもかまわないわけです。そうなると、何も LASTDRIVE コマンドも必要ないということになります。

このようなことを、CONFIG.SYS のカスタマイズの準備段階として実行しておかなければなりません。

使用する環境

メモリ環境を構築し、その効果を見るために基準となる環境を設定しておきましょう。

●使用する環境

ここでのテストにはIBMのThinkPad230Csを使っています。使用するCONFIG.SYSとAUTOEXEC.BATはオリジナルのものを使用しますが、MS-DOSのものはMS-DOS6.2/VをThinkPadにインストールした時に作成されるものです。

それらオリジナルは、あまり気持ちのいいものではありませんので、次に示すように編集してください。

PC-DOSの場合

…CONFIG.SYSのオリジナル…

```
BUFFERS = 20
FILES = 30
DOS = HIGH,UMB
COUNTRY = 081,932,C:¥DOS¥COUNTRY.SYS
SHELL = C:¥DOS¥COMMAND.COM □/P □/E:512
DEVICE = C:¥DOS¥$FONT.SYS
DEVICE = C:¥DOS¥HIMEM.SYS
DEVICE = C:¥DOS¥EMM386.EXE RAM □ X = C800-CFFF
DEVICEHIGH = C:¥DOS¥SETVER.EXE
DEVICEHIGH = C:¥DOS¥$DISP.SYS
DEVICEHIGH = C:¥DOS¥$IAS.SYS
DEVICE = C:¥EZGOING¥IBMDSS01.SYS □/M
DEVICE = C:¥EZGOING¥IBMDOSCS.SYS
DEVICE = C:¥EZGOING¥DICRMU01.SYS □/MA = C800-CFFF
DEVICE = C:¥EZGOING¥$ICPMDOS.SYS
```

```
DEVICE = C:¥DOS¥POWER.EXE
DEVICE = C:¥EZGOING¥AUTODRV.SYS C:¥EZGOING¥AUTODRV.INI
DEVICE = C:¥EZGOING¥DISKDRV.SYS
DEVICEHIGH = C:¥DOS¥$PRN.SYS
DEVICEHIGH = C:¥DOS¥ANSI.SYS /X
DEVICEHIGH = C:¥DOS¥$FDD12.SYS
REM DEVICEHIGH = C:¥DOS¥RAMDRIVE.SYS
INSTALL=C:¥DOS¥IBMMKKV.EXE /M=S /Z=4 /C /L /J=90 /S = C:¥DOS¥MULTDICT.PRO /U = C:¥$USRDICT.DCT
STACKS = 9,256
LASTDRIVE = Z
```

↓

…編集後の CONFIG.SYS…

```
BUFFERS = 20
FILES = 30
DOS = HIGH,UMB
COUNTRY = 081,932,C:¥DOS¥COUNTRY.SYS
SHELL = C:¥DOS¥COMMAND.COM /P C:¥DOS
DEVICE = C:¥DOS¥HIMEM.SYS
DEVICE = C:¥DOS¥EMM386.EXE RAM
DEVICEHIGH = C:¥DOS¥$FONT.SYS /P = C:¥FONT
DEVICEHIGH = C:¥DOS¥$DISP.SYS
DEVICEHIGH = C:¥DOS¥$IAS.SYS
DEVICEHIGH = C:¥DOS¥ANSI.SYS /X
INSTALL=C:¥DOS¥IBMMKKV.EXE /M=S /Z=4 /C /L /J=90 /S = C:¥DOS¥MULTDICT.PRO /U = C:¥$USRDICT.DCT
STACKS = 9,256
LASTDRIVE = Z
```

…AUTOEXEC.BAT のオリジナル…

```
@ECHO OFF
SET□COMSPEC = C:¥DOS¥COMMAND.COM
REM - 以下の3行のコマンドは初期導入用に設定しておりますので編
集時に
REM - 削除されても差し支えありません。
IF□NOT□EXIST C:¥F_IPL.COM□GOTO□SECOND
F_IPL
:SECOND
PROMPT□$P$G
PATH
C:¥WINDOWS;C:¥DOS;C:¥THINKPAD¥;C:¥IBMVESA;C:¥EZ-
GOING;C:¥LLRA;%PATH%;
SET TEMP = C:¥DOS
SET IBMAV = C:¥DOS
SET SPEECHDATA = C:¥SPEECHW¥BIN
C:¥THINKPAD¥FUELDOS
C:¥DOS¥SMARTDRV.EXE
LH□C:¥DOS¥KEYB.COM□JP,932,C:¥DOS¥KEYBOARD.SYS
LH□C:¥DOS¥PRNIBM.COM
REM□C:¥DOS¥MOUSE.COM
CALL□C:¥DOS¥IBMAVDR.BAT□C:¥DOS¥
C:¥LLRA¥LLRA1.EXE
VER
WIN
```

↓

…編集後の AUTOEXEC.BAT…

```
@ECHO OFF
SET□COMSPEC = C:¥DOS¥COMMAND.COM
```

```
PROMPT □$P$G
PATH □ C:¥;C:¥WINDOWS;C:¥DOS;
SET □ TEMP = C:¥DOS
C:¥DOS¥SMARTDRV.EXE
LH□C:¥DOS¥KEYB.COM□JP,932,C:¥DOS¥KEYBOARD.SYS
```

MS-DOSの場合

```
… CONFIG.SYS のオリジナル…
DEVICE = C:¥DOS¥SETVER.EXE
DEVICE = C:¥DOS¥BILING.SYS
DEVICE = C:¥DOS¥HIMEM.SYS
DOS = HIGH,UMB
DEVICE = C:¥DOS¥EMM386.EXE RAM
DEVICEHIGH = C:¥DOS¥JFONT.SYS □/P = C:¥DOS¥
DEVICEHIGH = C:¥DOS¥JDISP.SYS
DEVICEHIGH = C:¥DOS¥JKEYB.SYS □/106 □ C:
¥DOS¥JKEYBRD.SYS
DEVICEHIGH = C:¥DOS¥KKCFUNC.SYS
FILES = 30
DEVICEHIGH = C:¥DOS¥ANSI.SYS
DEVICEHIGH = C:¥DOS¥MSIMEK.SYS □/A1
DEVICEHIGH=C:¥DOS¥MSIME.SYS□/D＊□C:¥DOS¥MSIMER.
DIC /DC:¥DOS¥MSIME.DIC □/C1 □/N □/A1
                    ↓
…編集後の CONFIG.SYS …
FILES = 30
DOS = HIGH,RAM
DEVICE = C:¥DOS¥HIMEM.SYS
```

```
DEVICE = C:¥DOS¥EMM386.EXE RAM
DEVICEHIGH = C:¥DOS¥BILING.SYS
DEVICEHIGH = C:¥DOS¥JFONT.SYS □/P = C:¥DOS¥
DEVICEHIGH = C:¥DOS¥JDISP.SYS
DEVICEHIGH = C:¥DOS¥JKEYB.SYS □/106 □ C:
¥DOS¥JKEYBRD.SYS
DEVICEHIGH = C:¥DOS¥KKCFUNC.SYS
DEVICEHIGH = C:¥DOS¥ANSI.SYS
DEVICEHIGH = C:¥DOS¥MSIMEK.SYS □/A1
DEVICEHIGH=C:¥DOS¥MSIME.SYS□/D∗□C:¥DOS¥MSIMER.DIC
□/DC:¥DOS¥MSIME.DIC □/C1 □/N □/A1
```

…AUTOEXEC.BATのオリジナル…

```
C:¥DOS¥SMARTDRV.EXE □/X
@ECHO □ OFF
PROMPT □$p$g
PATH C:¥DOS
SET TEMP = C:¥DOS
C:¥DOS¥NLSFUNC.EXE □ C:¥DOS¥COUNTRY.SYS
C:¥DOS¥CHEV.COM JP
```

…編集後の AUTOEXEC.BAT…

```
C:¥DOS¥SMARTDRV.EXE □/X
@ECHO □ OFF
PROMPT □$p$g
PATH C:¥;C:¥WINDOWS;C:¥DOS
SET TEMP = C:¥DOS
C:¥DOS¥NLSFUNC.EXE □ C:¥DOS¥COUNTRY.SYS
C:¥DOS¥CHEV.COM □ JP
```

テストしているマシンの環境では、この設定で空きメモリが画面99と画面110のようになっています。

◎画面 99　空きメモリを確認する（PC-DOS）

```
C:¥>MEM

メモリーのタイプ        合計  =  使用済 +   使用可
----------------    -------   -------   -------
基本                  640K       72K      568K
UMB                    63K       58K        5K
アダプタ RAM/ROM       384K      384K        0K
XMS* メモリー        11,201K    3,842K    7,359K
----------------    -------   -------   -------
システム全メモリ      12,288K    4,356K    7,932K

1MB 以下全メモリ       703K      130K      573K

全 EMS メモリー                     11,584K (11,862,016 バイト)
使用可能 EMS* メモリー               7,552K  (7,733,248 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ           567K (581,040 バイト)
使用可能最大 UMB メモリー               4K   (4,080 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

◎画面 110　空きメモリを確認する（MS-DOS）

```
C:¥>MEM

メモリの種類          合計   =   使用   +   空き
----------------    -------   -------   -------
コンベンショナル          640K       80K      560K
上位                   62K       48K       14K
予約済み               384K      384K        0K
XMS メモリ*          11,202K    2,877K    8,325K
----------------    -------   -------   -------
全メモリ             12,288K    3,389K    8,899K

全 1MB 以下メモリ       702K      128K      574K

全 EMS メモリ                       11,584K  (11,862,016 バイト)
空き EMS メモリ*                     8,400K   (8,601,600 バイト)

* EMM386 は EMS メモリをシミュレートするために XMS メモリを使っています.
  空き EMS メモリは空き XMS メモリの変更によって変わることがあります.

最大実行可能プログラムサイズ              560K  (573,280 バイト)
最大空き上位メモリブロック                 10K   (10,224 バイト)
MS-DOS はハイメモリ領域に常駐しています.

C:¥>_
```

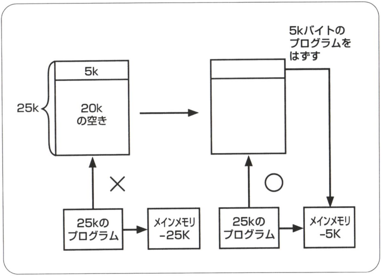
組み込まれる順番とサイズの関係

XMSメモリに組み込むことができますので、というのも、UMBの残りが少なくなってきた場合、その空き容量よりも大きいサイズの常駐容量のドライバや常駐型プログラムをロード使用とすると、そのすべてがUMBからはじき出されてコンベンショナルメモリにロードされてしまうからです。

たとえば、UMBの残りが20Kバイトである場合、常駐量が25Kバイトのデバイスドライバを DEVICEHIGH コマンドでロードしようとすると、その足りない5Kバイトのみがコンベンショナルメモリにロードされるのではなく、25Kバイトそっくりそのままコンベンショナルメモリにロードされてしまうのです。こうなるとコンベンショナルメモリは25K消費される上に、UMBは20Kバイト空いたままですから、もったいないことおびただしいですね。

もしも、そのドライバ以前に記述されているドライバで、必要な容量が5Kバイトのドライバがあったとしたらどうでしょう。その5Kバイトのドライバを25Kバイトのドライバと順番を入れ替えれば、うまいことUMBを利用できるという寸法です（図23)。

◎図23　順番を入れ替える



ただし、どのプログラムがどれくらいのメモリ量を必要とするかを調べるには、ちょっとした知識が必要です。

●プログラムの常駐量を知る

プログラムの常駐量はMEMコマンドなどでは知ることはできません。次の手法で

1．ファイルを調べる

これから実際に使用するメモリのカスタマイズソフトRAMBOOSTやMEMMAKERを利用してメモリを最適化すると、**RAMBOOST.INI**というファイル（C:¥DOS¥DATA¥）のLEAN.PIFセクションまたは**MEMMAKER.STS**（C:¥DOS¥）というファイルにその最適化に含まれているプログラムが必要とするメモリ量が記録されています（画面111、画面112）。

◎画面 111　LEAN.PIF

```
[learn.pif]

ramboost=      high,    need=176,     keep=176,                  data
ramboost=      high,    need=11808,   keep=11808,                data
$ias.sys=      high,    need=13616,   keep=6896,   size=13616,   code
ansi.sys=      high,    need=9552,    keep=4464,   size=9549,    code,      e
ms
files=         high,    need=1488,    keep=1488,                 data
fcbs=          high,    need=256,     keep=256,                  data
buffers=       high,    need=512,     keep=512,                  data
drives=        high,    need=2288,    keep=2288,                 data
stacks=        high,    need=624,     keep=624,                  data
stacks=        high,    need=2384,    keep=2384,                 data
ibmmkkv.exe=   high,    need=174224,  keep=22352,  size=180057,  code
command.com=   high,    need=65600,   keep=4320,   size=65091,   code,     em
s
smartdrv.exe=  high,    need=48608,   keep=29472,  size=48729,   code,     em
s
keyb.com=      high,    need=19728,   keep=6608,   size=19463,   code,     em
s
```

◎画面 112　MEMMAKER.STS

```
Command=C:¥DOS¥MSIME.SYS /D*C:¥DOS¥MSIMER.DIC /DC:¥DOS¥MSIME.DIC /C1 /N /A1
Line=16
FinalSize=8576
MaxSize=39888
FinalUpperSizes=0
MaxUpperSizes=0
ProgramType=DEVICE

Command=C:¥DOS¥SMARTDRV.EXE /X
Line=0
FinalSize=0
MaxSize=48992
FinalUpperSizes=29488
MaxUpperSizes=49552
ProgramType=PROGRAM

Command=C:¥DOS¥NLSFUNC.EXE C:¥DOS¥COUNTRY.SYS
Line=5
FinalSize=2544
MaxSize=8080
FinalUpperSizes=0
MaxUpperSizes=0
ProgramType=PROGRAM

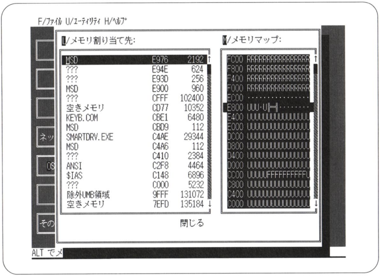
C:¥>_
```

2．MSDコマンドを使う

MSDの起動画面から **Alt** ＋ **U** → **M** を実行してメモリブロックを

表示させてみましょう（画面113）。左の「L/メモリ割り当て先」でメモリの使用量を調べたいプログラムを選択すると、そのプログラムが使用しているメモリ領域が右側の「M/メモリマップ」に表示されます。「‖」で囲まれた「=」の部分が割り当てられているメモリ領域です。この領域を計算することにより（一つの領域が1024バイト）、プログラムのメモリ必要量を知ることができます。

◎画面 113　メモリブロック



RAMBOOST

PC-DOS では RAMBOOST.EXE を使いますが、その起動ソフトとして RAMSETUP.COM があります。

● RAMBOOST を起動する

RAMBOOST をそのまま起動することもできますが、自分で各種オプションを指定しなければなりませんからやっかいです（画面114）。ここは一つ RAMSETUP の方を使用してみましょう。「**RAMSETUP ⏎**」とすれば RAMBOOST を対話形式で使用することができます（画面115）。

◎画面 114　RAMBOOST のオプション

```
C:¥>RAMBOOST

RAMBOOST - Central Point (R) Version 9.0
Copyright (c) Central Point Software, 1989-1993. All Rights Reserved.

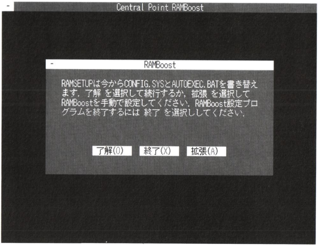
   [DEVICE|INSTALL=]RAMBOOST.EXE [パラメーター... ]

パラメーター:
  /?                   - このヘルプを表示する
  ACTIVE               - ACTIVE モードに設定する
  COMPLETE             - 初期プログラムのロードを完了
  DISABLE              - 構成ファイルを無効にする
  LEARN                - LEARN モードに設定する
  MODE                 - ｴﾗｰ･ﾚﾍﾞﾙ: 0=無効; 1=Activeﾓｰﾄﾞ; 3=Learnﾓｰﾄﾞ
  PIF                  - PIF情報の表示
  SYNC                 - 構成ファイルと適合させる
  TRACK [<ﾌｧｲﾙ>[/d]]   - LEARN モード時にファイルを追加／削除する

常駐のためのパラメーター:
  LOAD[=<ﾓｰﾄﾞ>]        - 常駐部分をメモリーに導入する
  /p=<ﾌﾟﾛﾌｧｲﾙ>         - 構成ファイルを指定する


C:¥>_
    英数 半角   R
```

◎画面 115　RAMSETUP で RAMBOOST を起動する



● RAMBOOST で設定を行う

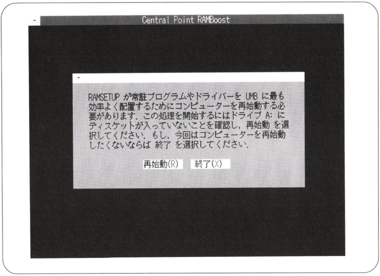
RAMSETUP の初期画面では「了解（O）」を選択してください。これでほぼ自動モードで設定を行うことができます。「拡張（A）」だと各設定を自分で行わなければなりませんので、より高度な使い方となります。

以後は次のような手順となります。

1．パソコンの再起動

初期画面で「了解（O）」を選択すると、画面116のようなメッセージが表示されます。もちろん「再始動（R）」を選択してください。

◎画面 116　パソコンがリセットされる



2．LEARN MODE

パソコンがリセットされると RAMBOOST が LEARN MODE で起動され、まずメモリ状況を把握し適切な設定を計算してくれます。

3．ACTIVE モード

再度パソコンがリセットされ、こんどは適切な設定でパソコンが起動されます。これで終わりです。

拍子抜けするほどあっけなく終わってしまいましたが、これでいいのです。早速 MEM コマンドでメモリ状況を見てみましょう（画面117）。

◎画面 117　MEM コマンドによるメモリ状況

```
C:¥>MEM

メモリーのタイプ     合計  =  使用済 +  使用可
----------------  -------   -------   -------
基本                640K       92K      548K
UMB                 168K      154K       13K
アダプタ RAM/ROM    384K      384K        0K
XMS* メモリー     11,096K    3,737K    7,359K
----------------  -------   -------   -------
システム全メモリ  12,288K    4,368K    7,920K

1MB 以下全メモリ    808K      247K      561K

全 EMS メモリー                     11,584K (11,862,016 バイト)
使用可能 EMS* メモリー               7,552K  (7,733,248 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ          547K (560,576 バイト)
使用可能最大 UMB メモリー              10K  (10,368 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

「お、すげぇ!!」と感嘆できるはずでしたが、結果はどうもあまり芳しくありませんね。逆に効率が悪くなっていて、利用できるメモリ量が減ってしまっています。

●変更箇所

実は、RAMBOOST で行う自動設定は完全ではないのです。というよりも、通常の環境においてはほとんどの場合 RAMBOOST を利用するとかえって利用できるメモリが減ってしまいます。どこが変更されたのでしょうか？　変更後の CONFIG.SYS と AUTOEXEC.BAT を確認しておきましょう（画面118、画面119）。

◎画面 118　RAMSETUP 後の CONFIG.SYS

```
C:¥>TYPE CONFIG.SYS
BUFFERS=20
FILES=30
dos=high
COUNTRY=081,932,C:¥DOS¥COUNTRY.SYS
SHELL=C:¥DOS¥COMMAND.COM /P C:¥DOS
DEVICE=C:¥DOS¥HIMEM.SYS
DEVICE=C:¥DOS¥EMM386.EXE RAM
device=c:¥dos¥ramboost.exe load
DEVICE=C:¥DOS¥$FONT.SYS /P=C:¥FONT
DEVICE=C:¥DOS¥$DISP.SYS /HS=OFF
DEVICE=C:¥DOS¥$IAS.SYS
DEVICE=C:¥DOS¥ANSI.SYS /X
INSTALL=C:¥DOS¥IBMMKKV.EXE /M=S /Z=4 /C /L /J=90 /S=C:¥DOS¥MULTDICT.PRO /U=C:¥$U
SRDICT.DCT
STACKS=9,256
LASTDRIVE=Z

C:¥>_



    英数 半角   R
```

◎画面 119　RAMSETUP 後の AUTOEXEC.BAT

```
C:¥>TYPE AUTOEXEC.BAT
@ECHO OFF
SET COMSPEC=C:¥DOS¥COMMAND.COM
PROMPT $P$G
PATH C:¥;C:¥WINDOWS;C:¥DOS;
SET TEMP=C:¥DOS
C:¥DOS¥SMARTDRV.EXE
c:¥dos¥keyb.com jp,932,c:¥dos¥keyboard.sys

C:¥>_



    英数 半角   R
```

変更箇所は次の通りです。

1．DOS コマンドのオプション

当初は「DOS = HIGH.UMB」となっていたものが、ここでは「DOS = HIGH」となっています。

2．DEVICEHIGH が DEVICE に

すべての DEVICEHIGH コマンドが DEVICE コマンドになっています。

3．RAMBOOST.EXE がロードされる

「DEVICE = C:¥DOS¥RAMBOOST.EXE LOAD」という一行が付け加えられています。これが RAMBOOST の正体です。このプログラムによって、さまざまなデバイスドライバや常駐プログラムを UMB にロードしたりすることができます。

4．LH が消えた

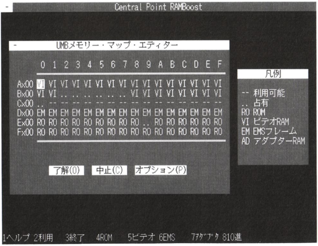
AUTOEXEC.BAT の変更点は、「LH」コマンドがなくなったことです。

ということは UMB がまったく利用されていないように感じられますが、そこはご心配なくちゃんと RAMBOOST がプログラムを UMB に配置してくれています。

●手動設定はどうか

RAMBOOST では初期画面から「拡張（A)」を選択して、手動で UMB の設定を行うことができます（画面120)。

◎画面 120　手動設定も可能



右の凡例に示されているように、現在使用できるUMBは「--」で示されています。しかし、この表示も正しいとは限りません。やはり信頼できるのはMEMコマンドです。

RAMBOOSTを利用する前と利用したあとのUMB状況を比較してみると、だいたいのところがわかるでしょう。

RAMBOOST前のUMBメモリーの詳細：

アドレス	領域	サイズ		名前	タイプ
0C800	1	5,248	(5K)	データ	
0C948	1	976	(1K)	IBMDOS	--空き -
0C985	1	10,528	(10K)	RAMBOOST	プログラム
0CC17	1	1,296	(1K)	IBMDOS	--空き -
0CC68	1	6,208	(6K)	$FONT	プログラム
0CDEC	1	45,152	(44K)	$DISP	プログラム
0D8F2	1	1,616	(2K)	$IAESKK	プログラム

```
0D957    1      272   (0K)            FCBS=4
0D968    1      640   (1K)            STACKS=9,128
0D990    1    1,248   (1K)            システム・スタック
0D9DE    1      112   (0K)   IBMDOS   -- 空き -
0D9E5    1    6,496   (6K)   KEYB     プログラム
0DB7B    1   18,512  (18K)   IBMDOS   -- 空き -
メモリーの概要：

メモリーのタイプ       サイズ  =     使用済  +    使用可能
----------------  ----------  ----------  ----------
基本                 655,360      33,024     622,336
UMB                   98,240      77,344      20,896
アダプタ RAM/ROM     393,216     393,216           0
XMS *メモリー     15,630,400   2,165,824  13,464,576
----------------  ----------  ----------  ----------
システム全メモリ  16,777,216   2,669,408  14,107,808

1MB以下全メモリ      753,600     110,368     643,232
```

RAMBOOST 後の UMB メモリーの詳細：

```
アドレス  領域      サイズ        名前       タイプ
--------  ----  ---------------  ---------  ----------------
0C800    1    5,248   (5K)              データ
0C948    1      976   (1K)   IBMDOS     -- 空き -
0C985    1   10,528  (10K)   RAMBOOST   プログラム
0CC17    1    1,296   (1K)   IBMDOS     -- 空き -
0CC68    1    6,208   (6K)   $FONT      プログラム
0CDEC    1   45,152  (44K)   $DISP      プログラム
0D8F2    1    1,616   (2K)   $IAESKK    プログラム
0D957    1      272   (0K)              FCBS=4
```

```
0D968     1        640   (1K)              STACKS=9,128
0D990     1      1,248   (1K)              システム・スタック
0D9DE     1        112   (0K)   IBMDOS     --空き -
0D9E5     1      6,496   (6K)   KEYB       プログラム
0DB7B     1     18,512  (18K)   IBMDOS     --空き -
メモリーの概要:

メモリーのタイプ         サイズ  =    使用済  +  使用可能
----------------   ----------  ----------  ----------
基本                  655,360      33,024     622,336
UMB                    98,240      77,344      20,896
アダプタ RAM/ROM      393,216     393,216           0
XMS *メモリー      15,630,400   2,165,824  13,464,576
----------------   ----------  ----------  ----------
システム全メモリ    16,777,216   2,669,408  14,107,808

1MB以下全メモリ       753,600     110,368     643,232
```

このようにメモリの使用状況を見ながら手動設定するのですが、その設定方法は千差万別でガイドラインを示すことさえもできません。いや、ガイドラインくらいは簡単ですが、そのガイドラインに沿って手動設定を行ったとしてもうまくいくとは限りません。へたに設定を行ってしまうとマシンが起動できないということもあるかもしれません。

●マルチ CONFIG の対応

RAMBOOST はマルチ CONFIG に対応していますので、CONFIG.SYS に複数の環境を記述しているユーザーでもそのまま RAMBOOST を利用することができます。

MEMMAKER

MS-DOSではMEMMAKERというメモリ管理・設定ユーティリティが附属しています。

● MEMMAKERを起動する

MEMMAKERは次のような形式で起動します。

MEMMAKER　[/B]　[/BATCH]　[/SESSION]　[/SWAP:ドライブ]　[/T]　[/UNDO]　[/W:サイズ1,サイズ2]

ここに見られるように、非常に多くのオプションを指定することができますが（画面121）、通常は何もオプションを指定しないで起動してもかまいません。

◎画面 121　MEMMAKERのオプション

```
MemMaker v1.00 - Memory Optimization Program
Copyright (c) 1992 Microsoft
All rights reserved.

Runs the MemMaker memory-optimization program.

MEMMAKER [/B][/BATCH][/SESSION][/SWAP:drive][/T][/UNDO][/W:size1,size2]

  /B              Displays MemMaker in black and white.
  /BATCH          Runs MemMaker in batch (unattended) mode.
  /BATCH2         Runs MemMaker in batch (unattended) mode and exit.
  /SESSION        Used by MemMaker during memory optimization.
  /SWAP:drive     Specifies the drive that was originally your startup
                  drive, if drive swapping is in effect.
  /T              Disables detection of IBM token-ring networks.
  /UNDO           Undoes MemMaker's most recent changes to your
                  startup files.
  /W:size1,size2  Specifies how much upper memory (in kilobytes) to
                  reserve for translation buffers.
                  Windows uses this space when running MS-DOS-based
                  applications.

C:¥>_
    英数 半角   R
```

このMEMMAKERの起動画面は、RAMSETUPのような華やかなものではありません。MEMMAKERの概要や実行方法が記述されていますので、とりあえず目を通してから⏎キーを押してください（画面122）。

◎画面 122　起動画面

```
Microsoft MemMaker

MemMaker へようこそ

MemMaker は，メモリ常駐プログラムやデバイスドライバを上位メモリ領域
へ移動して，システムのメモリを最適化します．これによって，アプリケー
ションが使えるコンベンショナルメモリがふえます．

MemMaker を実行した後は，メモリ常駐プログラムやデバイスドライバを追加
したり削除したりしない限り，コンピュータのメモリは最適化された状態で
す．メモリの設定を常に最適に保つためには，そのような変更を加えた後に
は，MemMaker を実行するようにしてください．

MemMaker では，選択できるオプションが反転表示のテキストで表示されま
す(たとえば，下の "続行" オプションを変えることができます)．
オプションの表示を変更するには，Space キーを押してください．選択した
いオプションが表示されたら，Enter キーを押してください．

MemMaker 実行中にヘルプを見るには，F1 キーを押してください．

    続行，それとも終了しますか? 続行

Enter=選択を変更しない  Space=選択を変更  F1=ﾍﾙﾌﾟ  F3=終了
```

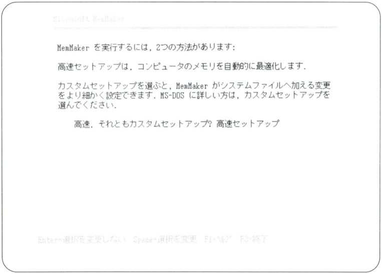
●設定を行う

MEMMAKERではほとんど自動的にメモリのカスタマイズを行うことができますが、次の手順で作業を進めます。

1．セットアップ方法を選択する

続いて表示される画面では、「カスタムセットアップ」または「高速セットアップ」を選択することができます（画面123）。最初は「高速セットアップ」を実行してください。

◎画面 123　セットアップ方法の選択

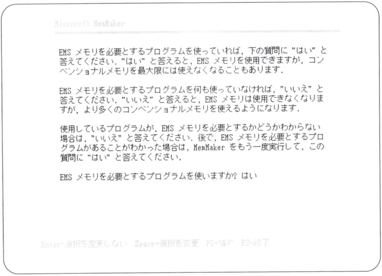


2．EMS を利用するプログラム

次に、EMS メモリを指定するプログラムを使うかどうかを指定してください（画面124）。

たとえば、ATOK8 などのかな漢字変換機能はほとんどが EMS 対応となっていますので、そのようなプログラムを使用する場合は「はい」を選択してください。EMS を使用するプログラムがあるかないか不明な場合も「はい」を選択しておいた方が無難です。

◎画面 124　EMS を使用するプログラムの指定



Microsoft MemMaker

EMS メモリを必要とするプログラムを使っていれば，下の質問に "はい" と答えてください．"はい" と答えると，EMS メモリを使用できますが，コンベンショナルメモリを最大限には使えなくなることもあります．

EMS メモリを必要とするプログラムを何も使っていなければ，"いいえ" と答えてください．"いいえ" と答えると，EMS メモリは使用できなくなりますが，より多くのコンベンショナルメモリを使えるようになります．

使用しているプログラムが，EMS メモリを必要とするかどうかわからない場合は，"いいえ" と答えてください．後で，EMS メモリを必要とするプログラムがあることがわかった場合は，MemMaker をもう一度実行して，この質問に "はい" と答えてください．

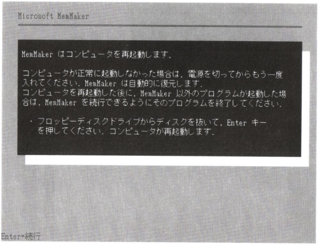
EMS メモリを必要とするプログラムを使いますか? はい

Enter=選択を変更しない　Space=選択を変更　F1=ﾍﾙﾌﾟ　F3=終了

3. リブート

MEMMAKER は Windows3.1 が組み込まれているかどうか検索したあと、マシンをリブートします。そのリブートが正常に行われなかった場合の対処方法などが表示されますので、読んでおいてください（画面125）。

◎画面 125　リブート



4．組み合わせの計算

マシンがリブートされると、デバイスドライバやプログラムをどの領域に配置するか、その組み合わせが計算されます。

計算が終了すると、最適と判断された環境でマシンをリブートします。注意事項が表示されますが、やはり正常にリブートできなかった場合の対処方法などです。

5．設定の確認

リブートされるとリブート後のシステム状況について確認する画面が表示されます。

ほとんどの場合問題なく起動されますので、ここでは「はい」を選択しておけばいいのですが、たまに素性の悪いパソコンや、周辺機器をやたら増設しているパソコンなどではうまく起動できないか怪しいメッセージが表示されることがあります。身に覚えのある方は、起動時の画面をよく見ておいてください。

6．設定前後のメモリ状況

最後に MEMMAKER の実行前と実行後とのメモリ状況が表示されます。その設定でよければ⏎キーを押すと終了します。気にいらない場合は

ESC キーを押すと、これまでの設定が取り消され MEMMAKER を実行する前の環境でマシンを起動することができます。

なお、MEMMAKER を起動する前にすでに最適なメモリ設定が行われている場合は、その旨が表示されます。

このように表示されたらもちろん新しい設定で起動する義理はありませんから、 F3 キーで新しい設定をキャンセルしてください。今回のサンプルですとこのメッセージが表示されてしまいました。

7．確認する

もしも新しい設定を受け入れた場合は、MEM コマンドで確認してみましょう。先ほどメモリ状況が表示されましたが、MEMMAKER を実行する前と後のメモリ状況がより一層明らかになります（画面126）。

◎画面 126　MEM コマンドで確認する

```
C:¥>MEM

メモリの種類        合計  =  使用  +  空き
----------------   -------  -------  -------
ｺﾝﾍﾞﾝｼｮﾅﾙ             640K      77K     563K
上位                  62K      51K      11K
予約済み              384K     384K       0K
XMS ﾒﾓﾘ*           11,202K   2,877K   8,325K
----------------   -------  -------  -------
全ﾒﾓﾘ              12,288K   3,389K   8,899K

全 1MB 以下ﾒﾓﾘ        702K     128K     574K

全 EMS ﾒﾓﾘ                           11,584K (11,862,016 ﾊﾞｲﾄ)
空き EMS ﾒﾓﾘ*                         8,400K  (8,601,600 ﾊﾞｲﾄ)

* EMM386 は EMS メモリをシミュレートするために XMS メモリを使っています.
  空き EMS メモリは空き XMS メモリの変更によって変わることがあります.

最大実行可能ﾌﾟﾛｸﾞﾗﾑｻｲｽﾞ                  562K (575,824 ﾊﾞｲﾄ)
最大空き上位ﾒﾓﾘﾌﾞﾛｯｸ                      7K   (7,488 ﾊﾞｲﾄ)
MS-DOS はハイメモリ領域に常駐しています.

C:¥>_
```

●カスタムセットアップ

カスタムセットアップを行うと、より一層空きメモリを確保することができます。

1．起動とカスタムセットアップの選択

MEMMAKER の起動画面では、「高速セットアップ」と「カスタムセ

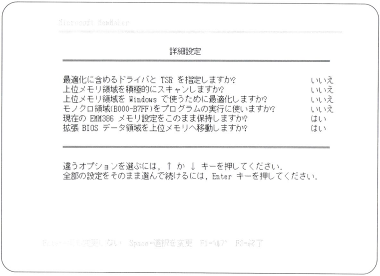
ットアップ」を選択することができました。デフォルトでは「高速セットアップ」となっていますが、 SPACE キーで「カスタムセットアップ」を選択することができます。

2．詳細の設定

「カスタムセットアップ」を選択し、EMSメモリを必要とするかしないかというメッセージに答えると、画面127のようにセットアップの詳細を設定できるようになります。

一番上の「最適化に含めるドライバと TSR を指定しますか？」と「上位メモリ領域を Windowsで使うために最適化しますか？」を「いいえ」に設定し、残りのすべてを「はい」としてみてください。

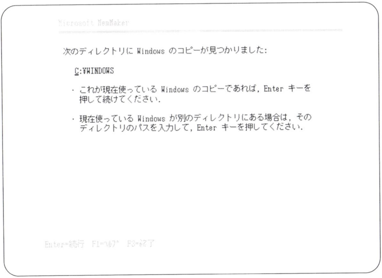
◎画面 127　詳細の設定



3．ウィンドウズディレクトリの指定

ウィンドウズのディレクトリが検索され表示されます (画面128)。このディレクトリでよければ⏎キーを押してください。他のディレクトリのウィンドウズを使用しているなら、そのディレクトリ名を入力してください。

◎画面 128　ウィンドウズのディレクトリを指定する



4．リブート

パソコンがリブートされますが、例によってその際の注意事項が表示されます。これは高速セットアップのものとほぼ同じです。

これ以後は高速セットアップの場合の「4．」以降と同じ手順となります。

5．確認

設定が終了したら MEM コマンドで確認してみましょう。コンベンショナルメモリは増えていませんが、UMB が相当増えています（画面129）。

◎画面 129　MEM コマンドで確認する

```
C:¥>MEM

メモリの種類        合計  =  使用  +  空き
----------------  -------  -------  -------
ｺﾝﾍﾞﾝｼｮﾅﾙ              640K      60K     580K
上位                  94K      68K      26K
予約済み              384K     384K       0K
XMS ﾒﾓﾘ*           11,170K   2,877K   8,293K
----------------  -------  -------  -------
全ﾒﾓﾘ              12,288K   3,389K   8,899K

全 1MB 以下ﾒﾓﾘ        734K     128K     606K

全 EMS ﾒﾓﾘ                        11,584K (11,862,016 ﾊﾞｲﾄ)
空き EMS ﾒﾓﾘ*                      8,368K  (8,568,832 ﾊﾞｲﾄ)

* EMM386 は EMS メモリをシミュレートするために XMS メモリを使っています.
  空き EMS メモリは空き XMS メモリの変更によって変わることがあります.

最大実行可能ﾌﾟﾛｸﾞﾗﾑｻｲｽﾞ                580K (594,112 ﾊﾞｲﾄ)
最大空き上位ﾒﾓﾘﾌﾞﾛｯｸ                   18K  (18,848 ﾊﾞｲﾄ)
MS-DOS はハイメモリ領域に常駐しています.

C:¥>_
```

6．Windows 用の設定

ここでの設定で**「B000-B7FF」**までも UMB に使ってしまうこともあります。この領域はモノクロ用のビデオ RAM ですが、ここを UMB として使ってしまうとウィンドウズが起動できなくなってしまうのです。

もちろん、その対策もあります。MS-DOS 附属の MONOUMB.386 というドライバをウィンドウズの SYSTEM.INI で組み込むのです。つまり、次の 1 行を **SYSTEM.INI の［386Enh］**セクションに記述します（画面 130）。

```
DEVICE = C:¥DOS¥MONOUMB.386 ↵
```

◎画面 130　SYSTEM.INI を編集する

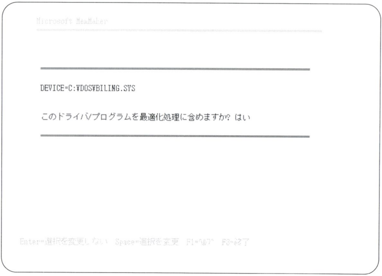
```
[386Enh]
SystemROMBreakPoint=false
EMMExclude=E000-EFFF
device=vdosvd.386
ebios=*ebios
display=vdd62x5j.386
EGA80WOA.FON=EGA80WOA.FON
EGA40WOA.FON=EGA40WOA.FON
CGA80WOA.FON=CGA80WOA.FON
CGA40WOA.FON=CGA40WOA.FON
DOSVFONTS=*18,12,8,8
DOSVWOA.FON=PS55SOEM.FON
device=C:¥WINDOWS¥SYSTEM¥WIN32S¥W32S.386
device=VSHARE.386
;======== PC DOS 6 Setup Modification - Begin ========
device=C:¥DOS¥SYSTEM¥CPBVXD.386
;======== PC DOS 6 Setup Modification - End ========
32BitDiskAccess=on
device=*int13
device=*wdctrl
mouse=*vmd
network=*VNETBIOS,VIPX.386,VCLIENT.386

C:¥WINDOWS¥SYSTEM.INI                    行    41   桁    1   挿入      E 3.12
F1=ﾎﾞｯｸｽ用文字                                    7=←ｼﾌﾄ  8=ｼﾌﾄ→      10=次ﾌｧｲﾙ
```

●デバイスドライバを指定する

詳細設定メニューで「**最適化**に含めるドライバと TSR を指定しますか？」で「はい」を選択すると、ドライバや常駐プログラムを一つひとつ選択しながら最適化の候補に加えることができます（画面131）。

◎画面 131　最適化候補の選択



どうしても最適化したくないドライバや、高速セットアップでは最適化されないドライバを強制的に最適化の候補に加えることができます。しかし、どのドライバ・常駐プログラムを選択すればいいのかはシステムによってまったく違いますし、選択するにはメモリやプログラムについての知識が必要ですから、特にこのオプションを指定する意義はありません。

●マルチ CONFIG には未対応

MEMMAKER は残念ながらマルチ CONFIG には対応していませんので、マルチ CONFIG を使用しているユーザーは、次の手順で MEMMAKER を利用してください。

1．それぞれの環境を作成する

環境ごとの CONFIG.SYS と AUTOEXEC.BAT を作成してください。

2．MEMMAKER でカスタマイズ

それぞれの環境でマシンを起動し、MEMMAKER でカスタマイズを行ってください。もちろん、カスタマイズした CONFIG.SYS と AUTOEXEC.BAT はすべて別の名前で保存しなければなりません。

3．マルチ CONFIG を作成する

すべてのCONFIG.SYSとAUTOEXEC.BATを一つのファイルにし(COPYコマンドを使えば簡単です)、それぞれを別々の環境として名前をつけます。

この場合、「COMMON」や「INCLUDE」は使ってはいけません。あくまでもこれまで作成した複数のCONFIG.SYSとAUTOEXEC.BATをつなげて名前を付けるだけです。

4．保存して起動

作成したCONFIG.SYSとAUTOEXEC.BATを保存してカスタマイズは終了です。

QEMMを使う

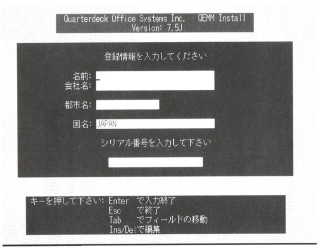
QEMMはDOS/Vパソコンでのメモリ管理ソフトの定番です。現在の最新バージョンは、7.5Jで完全に日本語化されています。

● QEMMのインストールとオプティマイズ

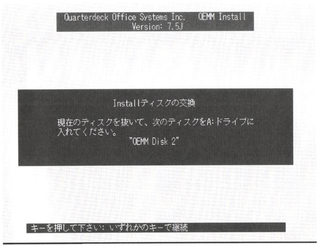
この**QEMM**は以前のバージョンに比べてさらに使いやすく簡単になりました。一応対話形式でインストールが進んでいきますが、ほとんどユーザーはお茶を飲みながら（パソコンの健康のため、たばこはやめましょう）リターンキーでも押していればいいのです。

必要な操作は、最初に行うユーザー登録と（画面132)、QEMMのシステムディスクを交換することくらいです（画面133)。また、**オプティマイズ**（最適化）の段階では、最適化の手段を選択する操作がありますが、ほぼこれだけです。

◎画面 132　ユーザー登録



◎画面 133　ディスクの交換



あとは、QEMM が勝手に必要なファイルをハードディスクに展開し、オプティマイズを行ってくれます。

オプティマイズの段階で「高速 OPTIMIZE」を選択すれば、あとはほぼ全自動です。QEMM が何度か自動的にマシンをリブートしたりします。システムによってはリセットスイッチでリブートしなければなりませんが、とにかく表示されるメッセージにしたがって操作すれば何も心配はいりません。

最後に新しい設定の詳細と、その設定を確定するかどうか選択できますので↵キーを押せばオプティマイズは完了です。

●どれくらいメモリが増えたか？

とりあえずフルオートモードで実行した結果を見てみましょう。もちろん MEM コマンドを使います（画面134、画面135）。

◎画面 134　PC-DOS

```
C:¥>MEM

メモリーのタイプ      合計  =  使用済 +  使用可
----------------   -------   -------   -------
基本                  640K       33K      607K
UMB                4,193,6   4,193,6        3K
アダプタ RAM/ROM      384K      384K        0K
XMS* メモリー       11,931K    4,491K    7,440K
----------------   -------   -------   -------
システム全メモリ    12,288K    4,238K    8,050K

1MB 以下全メモリ   4,194,2   4,193,6      610K

全 EMS メモリー                       11,520K (11,796,480 バイト)
使用可能 EMS* メモリー                 7,440K  (7,618,560 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ           607K (621,488 バイト)
使用可能最大 UMB メモリー                2K   (1,616 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

◎画面 135　MS-DOS

```
C:¥>MEM

メモリの種類        合計  =  使用  +   空き
----------------  -------  -------  -------
ｺﾝﾍﾞﾝｼｮﾅﾙ            640K      11K     629K
上位              4,193,8  4,193,7      93K
予約済み             384K     384K       0K
XMS ﾒﾓﾘ           11,758K   3,582K   8,176K
----------------  -------  -------  -------
全ﾒﾓﾘ             12,288K   3,390K   8,898K

全 1MB 以下ﾒﾓﾘ       147K  4,193,7     722K

全 EMS ﾒﾓﾘ                          11,408K (11,681,792 ﾊﾞｲﾄ)
空き EMS ﾒﾓﾘ                         8,176K  (8,372,224 ﾊﾞｲﾄ)

最大実行可能ﾌﾟﾛｸﾞﾗﾑｻｲｽﾞ                629K (643,776 ﾊﾞｲﾄ)
最大空き上位ﾒﾓﾘﾌﾞﾛｯｸ                   92K  (94,560 ﾊﾞｲﾄ)
MS-DOS はハイメモリ領域に常駐しています.

C:¥>_
```

PC-DOS では約 610K バイト、MS-DOS ではなんと一気に 630K バイト近くの空きメモリが確保されています。UMB も 90K バイトというすごさです。どんな CONFIG.SYS と AUTOEXEC.BAT でこのような空きメモリが確保されているのかも確認しておきましょう。

PC-DOS

…CONFIG.SYS…

```
BUFFERS = 20
FILES = 30
DOS = HIGH,UMB
COUNTRY = 081,932,C:¥DOS¥COUNTRY.SYS
SHELL = C:¥QEMM¥LOADHI.COM □/R:2 □ C:¥DOS¥COMMAND.COM □/P □ C:¥DOS
device = c:¥qemm¥dosdata.sys
DEVICE = C:¥QEMM¥QEMM386.SYS □ ST:M R:1 □ RAM
device = c:¥qemm¥dos-up.sys □@c:¥qemm¥dos-up.dat
```

```
DEVICE = C:¥QEMM¥LOADHI.SYS □/R:1 □/SIZE = 8656 □ C:
¥QEMM¥QDPMI.SYS □ SWAPFILE = DPMI.SWP □ SWAPSIZE =
1024
DEVICE = C:¥QEMM¥LOADHI.SYS □ / R:2 □ / SIZE = 26288 □ C:
¥DOS¥$FONT.SYS □/P = C:¥FONT
DEVICE = C:¥QEMM¥LOADHI.SYS □ / R:2 □ / SIZE = 27056 □ C:
¥DOS¥$DISP.SYS
DEVICE = C:¥QEMM¥LOADHI.SYS □/R:2 □/SIZE = 13904 C:
¥DOS¥$IAS.SYS
DEVICE = C:¥QEMM¥LOADHI.SYS □/R:2 □/SIZE = 9840 C:
¥DOS¥ANSI.SYS □/X
INSTALL=C:¥DOS¥IBMMKKV.EXE□/M=S□/Z=4□/C□/L□/J=
90 □/S = C:¥DOS¥MULTDICT.PRO □/U = C:¥$USRDICT.DCT
STACKS = 9,256
LASTDRIVE = Z

…AUTOEXEC.BAT…
@ECHO □ OFF
SET □ COMSPEC = C:¥DOS¥COMMAND.COM
PROMPT □$P$G
PATH □ C:¥QEMM;C:¥;C:¥WINDOWS;C:¥DOS;
SET □ TEMP = C:¥DOS
C:¥QEMM¥LOADHI □/R:2 □/LO □ C:¥DOS¥SMARTDRV.EXE
C:¥QEMM¥LOADHI □/R:2 □ C:¥DOS¥KEYB.COM □ JP,932,
C:¥DOS¥KEYBOARD.SY
```

MS-DOS

```
…CONFIG.SYS…
FILES = 30
DOS = HIGH,UMB
```

```
device = c:¥qemm¥dosdata.sys
DEVICE = C:¥QEMM¥QEMM386.SYS □ ST:M □ R:1 □ RAM
device = c:¥qemm¥dos-up.sys @c:¥qemm¥dos-up.dat
DEVICE = C:¥QEMM¥LOADHI.SYS □/R:2 □/SIZE = 8656 □ C:
¥QEMM¥QDPMI.SYS □ SWAPFILE = DPMI.SWP □ SWAPSIZE =
1024
DEVICE=C:¥QEMM¥LOADHI.SYS □ / R:1 □ / SIZE=12752 □ C:
¥DOS¥BILING.SYS
DEVICE=C:¥QEMM¥LOADHI.SYS □ / R:2 □ / SIZE=22624 □ C:
¥DOS¥JFONT.SYS □/P = C:¥DOS¥
DEVICE=C:¥QEMM¥LOADHI.SYS □ / R:2 □ / SIZE=21888 □ C:
¥DOS¥JDISP.SYS
DEVICE=C:¥QEMM¥LOADHI.SYS □ / R:2 □ / SIZE=17472 □ C:
¥DOS¥JKEYB.SYS □/106 □ C:¥DOS¥JKEYBRD.SYS
DEVICE = C:¥QEMM¥LOADHI.SYS □/R:2 □/SIZE = 4960 □ C:
¥DOS¥KKCFUNC.SYS
DEVICE = C:¥QEMM¥LOADHI.SYS □/R:2 □/SIZE = 9968 □ C:
¥DOS¥ANSI.SYS
DEVICE=C:¥QEMM¥LOADHI.SYS □ / R:2 □ / SIZE=65456 □ C:
¥DOS¥MSIMEK.SYS □/A1
DEVICE=C:¥QEMM¥LOADHI.SYS □ / R:2 □ / SIZE=40176 □ C:
¥DOS¥MSIME.SYS □/D * □ C:¥DOS¥MSIMER.DIC □/DC:
¥DOS¥MSIME.DIC □/C1 □/N □/A1
SHELL=C:¥QEMM¥LOADHI.COM□/R:2□C:¥COMMAND.COM
C:¥□/P

…AUTOEXEC.BAT…
C:¥QEMM¥LOADHI□/R:2□/LO□C:¥DOS¥SMARTDRV.EXE□/X
@ECHO OFF
PROMPT □$p$g
PATH C:¥QEMM;C:¥DOS
```

```
SET □ TEMP = C:¥DOS
C:¥QEMM¥LOADHI □/R:2 □ C:¥DOS¥NLSFUNC.EXE □ C:
¥DOS¥COUNTRY.SYS
C:¥DOS¥CHEV.COM JP
```

● MANIFEST は便利

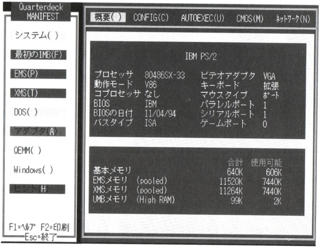
QEMM にはさまざまなユーティリティが附属していますが、その中でも「MANIFEST」と呼ばれるユーティリティは非常にありがたいものです。

この MANIFEST は QEMM をインストールすると自動的にインストールされますし、QEMM のディレクトリにはパスが通されていますので、どこのディレクトリからでも **「MFT ↵」** とすれば MANIFEST を起動することができます。

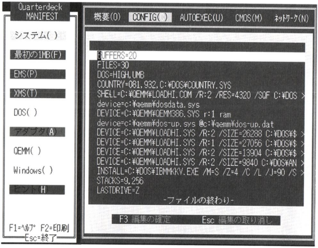
システム(S)

起動された時点では、システム情報が表示されています（画面136)。それればかりでなく、CONFIG.SYS と AUTOEXEC.BAT の編集も行うことができますので（画面137)、なにか問題を発見したときにスピーディに CONFIG.SYS を編集することができます。

◎画面 136　システム情報



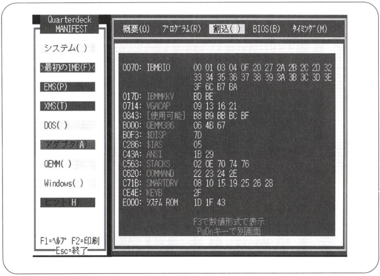
◎画面 137　CONFIG.SYS も編集することができる



最初の1MB(F)

通常 DOS が管理している 1MB のメモリ状況および、割り込みベクタの詳細を表示する機能など（画面138）、MEM コマンドの「/C」オプションと「/D」オプションを同時に使ったようなものです。

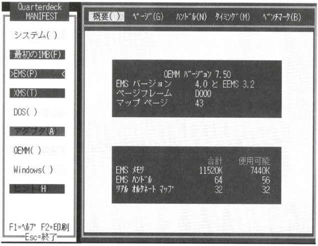
◎画面 138　割込(I)



EMS(P)

EMS についての情報が表示されます（画面139）。ここでは、EMS のバージョンやらページフレームアドレスなどが表示されていますので、EMS と他のプログラムやハードウェアがメモリのコンフリクトを起こしたときにページフレームアドレスを調べたり、ハンドル名を確認したりすることができます。

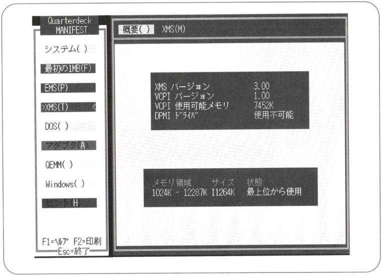
◎画面 139　EMS についての情報



XMS(T)

XMS の概要と詳細を見ることができますが（画面140）、XMS は直接ユーザーがどうのこうのいうところではありませんので、利用方法によってはそこそこ役立つかなというところです。

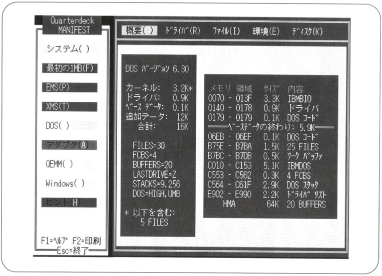
◎画面 140　XMS についての情報



DOS(D)

DOSのバージョンをはじめとする各種情報、DOSの標準ドライバやCONFIG.SYSで組み込まれているドライバの情報などを知ることができます（画面141）。また、環境変数のサイズや残りの使用可能なバイト数も表示されますので、CONFIG.SYSの/Eオプションで環境変数領域を指定する場合に役にたちます。

◎画面 141　DOS についての情報

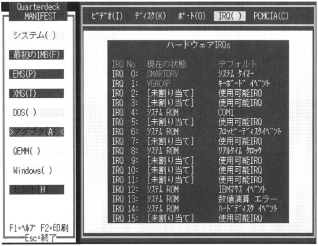


アダプタ(A)

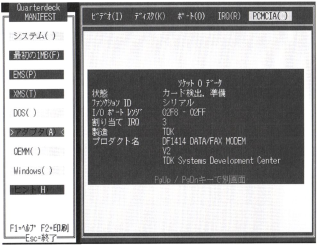
周辺機器の接続情報とその詳細を知ることができます。とくに「IRQ(R)」は、IRQのコンフリクトを調べたりするのに非常に役にたちます(画面142)。

また、ソケットサービス、カードサービスが組み込まれていれば、その状況を表示したり、実際にセットされている PCMCIA カードのデータを取得することもできます (画面143)。

◎画面 142　IRQ の割り当て状況も確認することができる



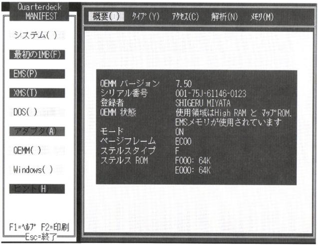
◎画面 143　PCMCIA カードの情報も見ることができる



QEMM(Q)

QEMMの状況や、メモリマップなどメモリ関係の情報が満載です（画面144）。メモリ関連の状況がある程度把握できますので、メモリのカスタマイズする際の参考となります。

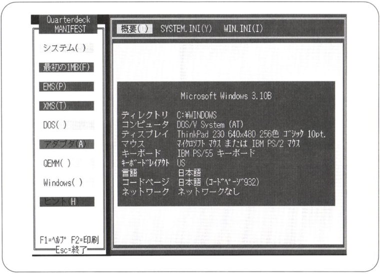
◎画面144　QEMMそのものの情報



Windows(W)

インストールされているWindows3.1についての情報も表示させることができます（画面145）。

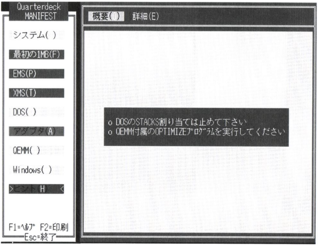
◎画面 145　Windowsについての情報



ヒント(H)

役立つ機能が満載のMFTの中でも、とくにこのヒントは重宝します(画面146)。何しろ、現在の設定の問題点を表示してくれますので、その問題点を解決すればシステムにとって最高の環境（メモリ環境）ということになります。

◎画面 146　なんとヒントが



ここで示された対策のうち、AUTOEXEC.BATやCONFIG.SYSを編集するだけで済むのでしたら、MFTの「システム(S)」で編集できますので、はやいとこ編集しておきましょう。

なお、STACKSコマンドで指定していないにも関わらず、「DOSのSTACKS割り当ては止めてください」というメッセージが表示されることがあります。これは、STACKSのデフォルトが「8,128」となっているためです。

このエラーを修正するには、CONFIG.SYSに次のように指定してください。

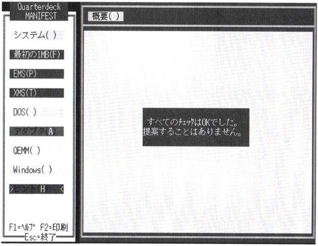
```
STACKS = 0,0 ↵
```

修正を行ったらマシンをリブートし、再度MFTの「ヒント(H)」を見てください。そこで「QEMM附属のOPTIMIZEプログラムを実行してください」というように表示されていれば、OPTIMIZEを実行してください。

現在の環境で何も問題がなければ画面147のようなメッセージが表示されます。このメッセージが表示されるまで修正とOPTIMIZEを繰り返して

ください。

◎画面 147　何も問題がない場合のメッセージ



なお、組み込むデバイスドライバが少ないうちはOPTIMIZEするのにあまり時間はかからないのですが、だんだん多くなってくると非常に時間がかかります。時には１時間以上かかることもありますが、数百億（あるいは数千億）の組み合わせを計算し最適な組み合わせを探すのですから、時間がかかるのもしかたのないことです。のんびりできるときにOPTIMIZEしましょう。

●エラーを起こした場合

何らかの原因でOPTIMIZEがうまくいかなかったり、OPTIMIZEはうまくいってもWindows3.1が起動できなくなったりすることがあります。このような場合は次の手順でエラー原因を見つけてください。

1．QWINFIXを実行する

QEMMにはQWINFIX.COMというプログラムが付属しています。このプログラムを実行してみてください。

2．ステルス機能を削除する

CONFIG.SYSの「QEMM386.SYS」の行に「ST:M」または「ST:F」というパラメータがあればこれを削除してください。

3．DOSDATA.SYSとDOS-UP.SYSを削除する

以下のような2行があったら、まずDOSDATA.SYSの行を削除し、それでもだめならその2行を削除してみましょう。

```
DEVICE = C:¥QEMM¥DOSDATA.SYS
DEVICE = C:¥QEMM¥DOS-UP.SYS □@□ C:¥QEMM¥DOS-UP.
DAT
```

以上のような処理は、誰でも簡単にできるものです。もしもこの処理を行ってもうまくWindows3.1やパソコンそのものが動作しない場合は、少々やっかいです。

●以上の処理でうまくいかない場合

これまでの処理でうまくいかない場合は、根本的にシステムを再構築しなければなりません。

1．初心に戻る

そのマシンのオリジナルのCONFIG.SYSとAUTOEXEC.BATに対してOPTIMIZEを実行してください。この時点ですでに動作がおかしい場合は、周辺機器やパソコン本体の問題がほとんどです。

2．必要なものから

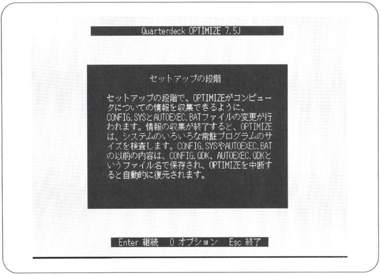
周辺機器の中で必要なものからデバイスドライバを組み込み、OPTIMIZEしていきます。この方法でどんどん組み込みを行い、最初にひっかかったデバイスドライバが悪さをしていることとなります。

と、文章にすると非常に簡単に聞こえますが、実際はいばらの道です。気長にやる覚悟が必要です。

●カスタム OPTIMIZE を行う

QEMM の OPTIMIZE には「カスタム OPTIMIZE」というメニューがありますが、この OPTIMIZE 方法は OPTIMIZE の各段階を監視することができるというオプションで、内容は高速セットアップとあまりかわりはありません。せいぜい出だしで処理を選択できるくらいです（画面148)。

◎画面 148　カスタム OPTIMIZE



そもそも、QEMM のオプション自体非常にたくさんあって、それをすべて解説しどのように利用するかという話になると本書一冊くらいの分量にはなるかもしれません。また、それらを完全に活用するには、メモリについてほとんどプロ級の知識が必要です。

QEMM の高速カスタマイズは非常に進化し、それだけでも十分用をなすものですから高速カスタマイズにまかせておいて、問題点を解決するくらいの方がいいでしょう。

●パソコンが起動できなくなった場合

QEMMのOPTIMIZEでエラーが起きると、パソコン自体が起動できなくなることもあります。その場合は、起動時に F5 を押してCONFIG.SYSとAUTOEXEC.BATをスキップしてしまえばいいのですが、QEMMが組み込まれているとQEMMのクイックブート機能でブートされ、マシンによっては F5 が効かないことあります。その場合は F8 で全てのCONFIG.SYSとAUTOEXEC.BATの行をスキップして起動してください。

OPTIMIZEを実行する前のオリジナルのCONFIG.SYSとAUTOEXEC.BATに戻してから再度起動してください。

MEMORY SERVER IIを使う

I/Oデータから「MEMORY SERVER II」というメモリ管理ソフトが発売されています。もともとNECのPC-98シリーズ用のメモリ管理ソフトとして有名な「MEMORY SERVER II」ですが、DOS/V用として新たにお目見えしました。その実力は立証済みです。

●インストールと設定

「MEMORY SERVER II」にもインストーラが附属していますが、このインストーラで必要なファイルのコピーからメモリ環境の設定まですべてをこなすことができるようになっています。もちろん最初は「フルオートインストール」です。

インストーラは、「MEMORY SERVER II」のシステムディスクのドライブ名と、組込先のドライブ名を指定して起動します。たとえば、システムディスクをAドライブにセットし、Cドライブのハードディスクに組み込みたいなら次のようにしてください。

```
A:¥INSTALL □ A:□ C:↵
```

以後、次のような手順でインストールと設定を行っていくこととなっています。

1．インストール先の確認

必要なファイルのコピー先や更新されるファイルが表示されますが、ここでは確認だけなので画面に表示されている内容でよければ↵キーを、インストールを中止するなら ESC キーを押してください。

2．ファイルのコピー

いきなりファイルのコピーが始まり、CONFIG.SYS と AUTOEXEC.BAT を更新してくれます（古いものはそれぞれ拡張子を.000 として保存されます）。これで組み込み終了です。

3．再起動

画面の指示にしたがって、フロッピーディスクを抜きマシンをリブートすると新しく構築された環境でマシンが起動されます。

4．確認する

起動されたら、早速 MEM コマンドでメモリ状況を確認してみましょう(画面149)。

◎画面 149　MEM コマンドで確認する

```
C:¥>MEM

メモリーのタイプ      合計  =  使用済 +  使用可
----------------  -------   -------   -------
基本                 640K      73K      567K
UMB                   64K      57K       7K
アダプタ RAM/ROM     384K     384K        0K
XMS* メモリー     11,200K   3,888K    7,312K
----------------  -------   -------   -------
システム全メモリ  12,288K   4,402K    7,886K

1MB 以下全メモリ     704K     130K      574K

全 EMS メモリー                  10,960K (11,223,040 バイト)
使用可能 EMS* メモリー            7,568K  (7,749,632 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ        567K (580,208 バイト)
使用可能最大 UMB メモリー             7K   (6,752 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

UMB は増えていますが、基本メモリは逆に若干減ってしまっています。このサンプルが UMB を利用するようなものがあまりないので、こうなっているのかもしれません。

●手動でインストールする

「MEMORY SERVER II」起動時に「カスタムインストール」を選択すると、細かく設定を行いながらインストールすることができます。

1．新規登録

ここでは初めての登録ですから「新規登録」を選択してください。デフォルトの設定が表示されます。

2．設定を行う

■ サーバの選択

「サーバ」というのは簡単に言えば、どのようなメモリ規格を利用するのかということです。ここで選択できるのは、「XMS」「EMS」「VCPI」（コラム参照）の3つです。

デフォルトではすべてを選択するとなっています。特にVCPIは必要ありませんが、マニュアルによるとPlug&Playドライバを利用する時に必要だということです。

■ UMBの使用

もちろん「する」を選択してください。

■ 起動時の情報表示

当初はここも「する」にしてください。

なお、ここで PageUp キーを押すと、さらに細かく設定することができるようになりますが、ここもとくにデフォルトの設定を変更する必要はありません。

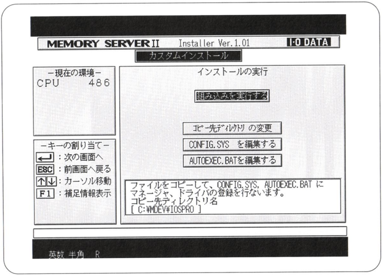
ただし、ここでEMSのページフレームを自動検出する設定にしておくと、EMSはC000からとられますので、PCMCIAカードを利用するユーザーで、カードに割り当てる領域をC000～CFFFに設定しているユーザーは、その領域を変更するか、ここでEMSのページフレームがD000から始まるように指定しなければなりません。

3．インストールの実行

設定が終わったら「次の画面へ」を選択すると、インストールの実行画面

となります（画面150）。

◎画面 150　インストールの実行画面



ここではCONFIG.SYS、AUTOEXEC.BATを編集したり、コピー先ディレクトリを変更したりすることができますが、CONFIG.SYSやAUTOEXEC.BATをこの段階で編集できるのは一部のエキスパートでしょうから、ここで何かするとしたら「コピー先ディレクトリの変更」くらいでしょう。それも必要ないのなら、「組み込みを実行する」を選択してインストールを実行しましょう。

あとはフルオートインストールと同じです。

4．確認

インストールが終了したら、例によってマシンをリブートしてからMEMコマンドでメモリを確認しましょう（画面151）。

◎画面 151　MEM コマンドで確認する

```
C:¥>MEM

メモリーのタイプ     合計  =  使用済 +  使用可
----------------  -------   -------   -------
基本                 640K       73K      567K
UMB                   64K       57K        7K
アダプタ RAM/ROM     384K      384K        0K
XMS* メモリー     11,200K    3,888K    7,312K
----------------  -------   -------   -------
システム全メモリ  12,288K    4,402K    7,886K

1MB 以下全メモリ     704K      130K      574K

全 EMS メモリー                  10,960K (11,223,040 バイト)
使用可能 EMS* メモリー            7,568K  (7,749,632 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ        567K (580,208 バイト)
使用可能最大 UMB メモリー             7K   (6,752 バイト)
DOS は HMA 領域に常駐しています

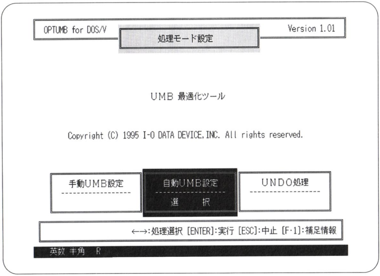
C:¥>_
    英数 半角   R
```

せっかく手動で設定しても、フルオートでの設定とまったく同じです。カスタムインストールといっても、そうシビアにメモリを使うような設定はできませんので、いたしかたのないことです。

● UMB をカスタマイズする

さて、もっとメモリを確保する手だてはないかと探すと、ありましたありました。「OPTUMB.EXE」というプログラムです。実行してみましょう(画面152)。

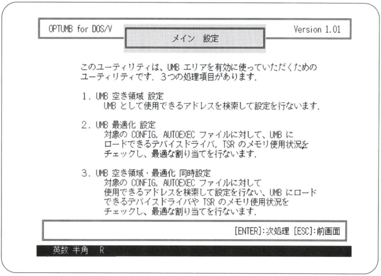
◎画面152　OPTUMB起動画面



1．処理モードを選択する

ここでも「手動」「自動」という処理モードを選択することができますが、「手動」ではやはり相当メモリについての知識と経験が必要とされますので(画面153)、完全に設定を行う自信がある方以外は「自動」を選択した方が無難です。

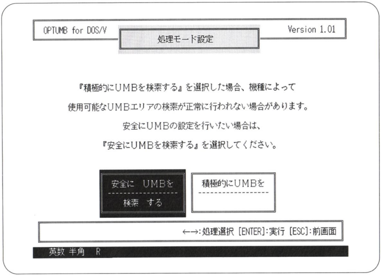
◎画面 153　手動モードでの設定



2．さらにモード設定

自動モードを選択すると、さらに「安全」でいくか「積極的」に UMB を活用するかを選択しなければなりません（画面154)。

◎画面 154　さらにモード選択

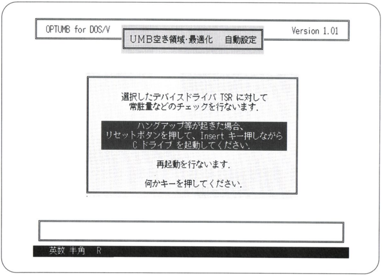


なるべく広いエリアを確保するため「積極的」を選択しましょう。

3．設定の実行

この設定では、マシンが3回リセットされます。ハングアップ時の対応も表示されますのでよく呼んでおいてください（画面155)。

◎画面155　ハングアップ時の対処方が表示される



4．設定の確認

自動的に設定された内容を確認することができます。もちろん変更することもできますが、不適切な設定を行おうとするとオンラインヘルプで怒られます。せっかくの自動モードですからそのまま次のステップへ進みましょう。

5．確認

これであとはおまかせです。カスタマイズが終了したらMEMコマンドで確認してみましょう（画面156）。

◎画面 156　MEM コマンドで確認する

```
C:¥>MEM

メモリーのタイプ     合計  =  使用済 +  使用可
----------------  -------  -------  -------
基本                640K      59K     581K
UMB                  92K      72K      20K
アダプタ RAM/ROM    384K     384K       0K
XMS* メモリー    11,172K   3,876K   7,296K
----------------  -------  -------  -------
システム全メモリ 12,288K   4,391K   7,897K

1MB 以下全メモリ    732K     131K     601K

全 EMS メモリー                  10,944K (11,206,656 バイト)
使用可能 EMS* メモリー            7,552K (7,733,248 バイト)

* EMM386はEMSメモリーを提供するため必要に応じてXMSメモリーを使用します。
  EMSメモリーとXMSメモリーの使用可能サイズは対応して変わることがあります。

実行可能プログラム最大サイズ        581K (594,864 バイト)
使用可能最大 UMB メモリー            18K  (18,128 バイト)
DOS は HMA 領域に常駐しています

C:¥>_
    英数 半角   R
```

劇的な成功とはいかないまでも、ある程度メモリが増えています。しかも、UMB にまだまだ余裕があるので、UMB にロードできるドライバや常駐ソフトなどをインストールしても、ほとんどコンベンショナルメモリには影響を及ぼしません。

INDEX

[記号&数字]

[A～Z]

●著者略歴●
宮田　茂（みやた　しげる）
1955年東京生まれ。
慶応義塾大学法学部法律学科卒業
現在、株式会社　シーガーズ　代表取締役

主な著書
ハードディスクのダイエット（広文社）
イメージスキャナ活用ハンドブック（新紀元社：共著）
DOS/V オンラインソフト集（インプレス）
パソコンこれから始める人が読む本（三笠書房）
NIFTY-Serve いきなり達人宣言（エーアイ出版）
その他各パソコン雑誌などに執筆中

■テクニカルライター募集
コンピュータ関連の単行本を執筆できるライターを募集しています。キャリアは問いません。我こそはと思われる方はお気軽にお問い合わせください。
TEL　(5295)8106
info @ Kobunsha.co.jp
■ご質問について
本書の内容についてのご質問は、ファクシミリ、お手紙でお寄せください。
FAX　03(3258)0488

PC/AT 互換機対応　CONFIG. SYS がわかる本

1995年9月1日　初版発行
1997年8月1日　第13刷発行

著　者　宮田　茂
発行者　七海典男

〒101　東京都千代田区神田東松下町28
電話(03)3258-0567(代表)
(03)3258-0488(FAX)
郵便振替00150-6-114868
株式会社広文社

印刷　三松堂印刷株式会社
製本　根本製本株式会社
ISBN 4-905999-16-2　C2055

ISBN4-905999-16-2
C2055 P2000E

9784905999164

定価2000円（本体1942円・税58円）
（2269）広文社

1912055020003